EDAM-8000
DIO series

Data Acquisition Modules
User's Manual

Web site: www.inlog.com.tw

Edition 1.12 **Date: 07/05/2011**

Table of Contents

Chapter 1 Introduction

1.1 Overview

The *EDAM-8000* DIO modules is a set of intelligent sensor to computer interface modules containing built-in microprocessor. They provide data comparison, and digital communication functions. Some modules provide digital I/O lines for controlling relays and TTL devices.

1.2 Module Compatibility

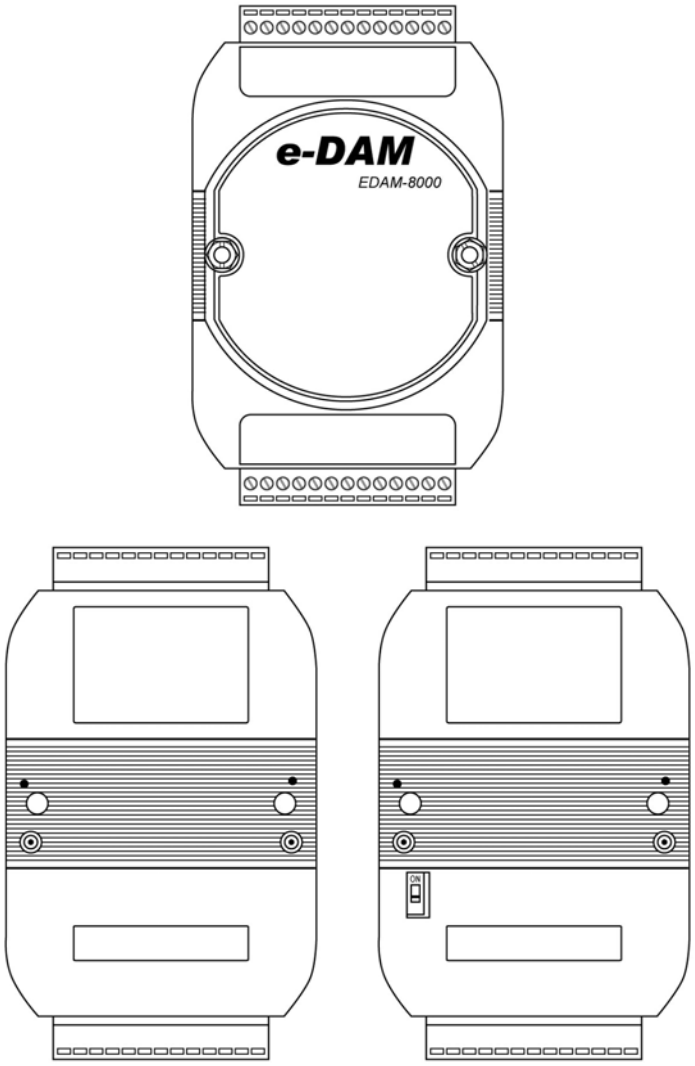
The *EDAM-8000* series are fully compatible to Advantech® ADAM-4000 series, ADlink® NμDAM-6000 series and ICP® I-7000 series .

1.3 Communication and Programming

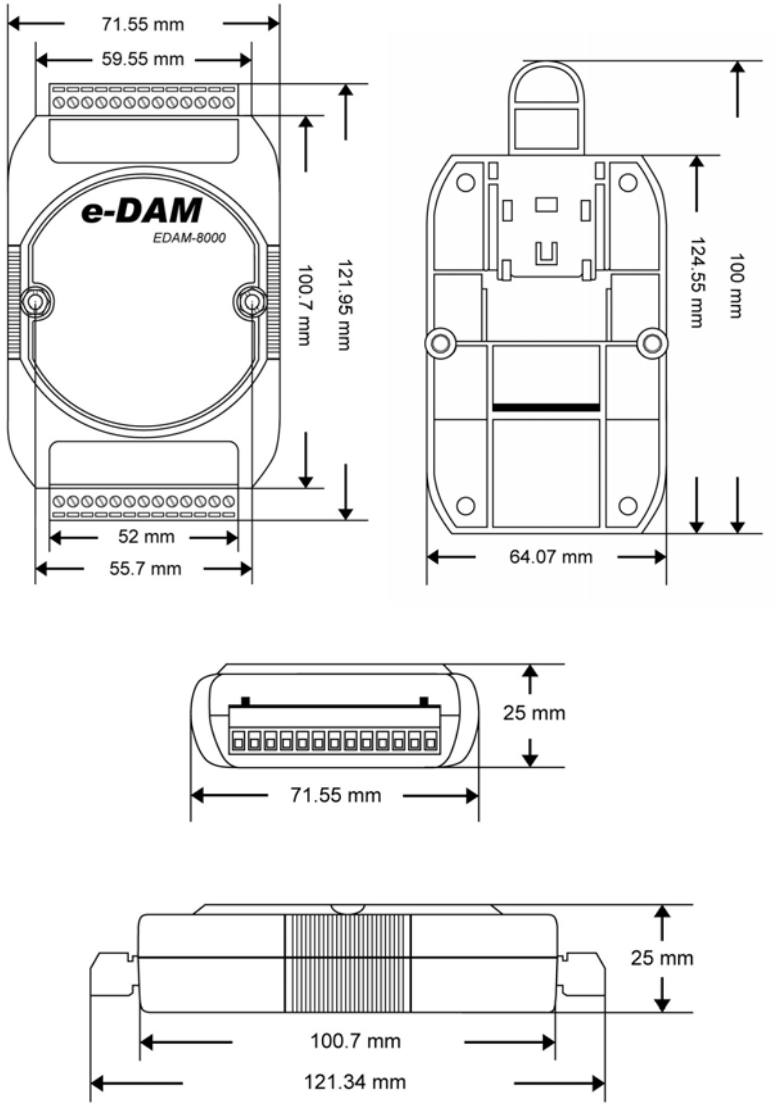
EDAM modules can connect to and communicate with all computers and terminals. They use RS-485 transmission standards, and communicate with **EDAM-ASCII** format protocol(default) or **Modbus-RTU** protocol(for firmware version D02.01 and later), which means that EDAM modules can be programmed in virtually any high-level language. Up to 256 EDAM modules may be connected to an RS-485 multi-drop network by using the EDAM RS-485 repeater, extending the maximum communication distance to 4,000 ft.

Chapter 2 About the EDAM DIO Modules

2.1 Outline of EDAM DIO modules



2.2 Module Dimension



2.3 Summary of modules

The EDAM provides a series of digital input or output modules to sense the digital signal or to control the remote devices.

Communication modules	
Module	Description
8520A	Isolated RS-232 to RS-422/485 converter
8510	Isolated RS-422/485 repeater
8531	Isolated USB to RS-485 & RS-422(TX,RX,RTS,CTS) Converter
8530	Isolated USB to RS-232 / RS-422 / RS-485 Converter

DC Input modules		
Module	Input channels	Input type
8041(D)	14	Isolated single ended with common source
8051(D)	16	Isolated single ended with common scource or common ground
8052(D)	8	Isolated with 8 differential input(sink/source)
8053(D)	16	Non-isolated single ended input

DC Output modules		
Module	Output channels	Output type
8042(D)	13	Isolated Open collector (NPN) with common power
8043(D)	16	Non-Isolated Open collector (NPN)
8045(D)	16	Isolated with open source (N-MOSFET)

DC Input and DC Output modules				
Module	Input ch.	Input type	Output ch.	Output type
8044(D)	4	Isolation with common source	8	Isolation with Open collector (NPN)
8050(D)	8	Non-isolated input channels(sink)	8	Non-Isolated with Open collector (NPN)
8050A(D)	8	Non-isolated input channels(source)	8	Non-Isolated with Open collector (PNP)
8055(D)	8	Isolated single ended with common source	8	Isolated with open source (N-MOSFET)

AC Input modules		
Module	Input ch.	Input type
8058(D)	8	Isolation with differential AC input (max. 250VAC)
8059(D)	8	Isolation with differential AC input (max. 80VAC)

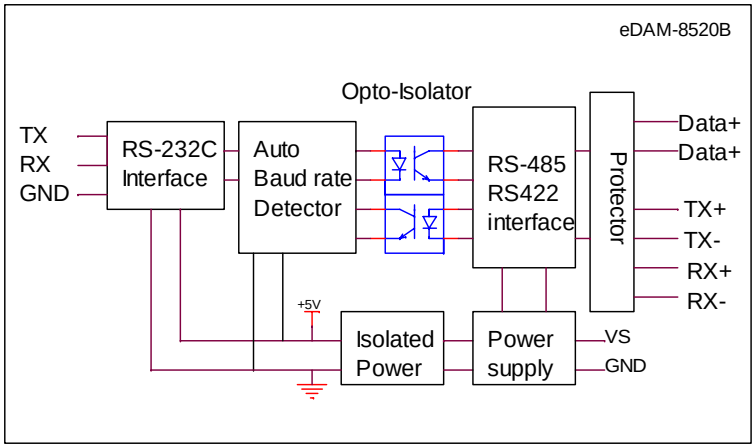
Relay Output and DC Input modules					
Module	Output ch.	Output type	Contact rating	Input ch.	Input type
8060(D)	4	RL1,RL2 Form A RL3,RL4 Form C	0.6A@125VAC 2A@30VDC	4	Isolation with common source
8063(D)	3	Form A	5A@250VAC 5A@30VDC	8	Isolation with common source
8065(D)	5	Form A	5A@250VAC 5A@30VDC	5	Isolation with common source
8067/F (D)	8	Form A	0.5A@120VAC 1.0A@24VDC	No input	

Solid-State Relay Output and DC Input modules					
Module	Output ch.	Output type	Contact rating	Input ch.	Input type
8063A(D)	3	ACC-SSR, Normal Open	24~265Vrms, 1.0Arms	8	Isolation with common source
8065A(D)	5	ACC-SSR, Normal Open	24~265Vrms, 1.0Arms	4	Isolation with common source
8063B(D)	3	DC-SSR, Normal Open	3~30VDC, 1.0A	8	Isolation with common source
8065B(D)	5	DC-SSR, Normal Open	3~30VDC, 1.0A	4	Isolation with common source

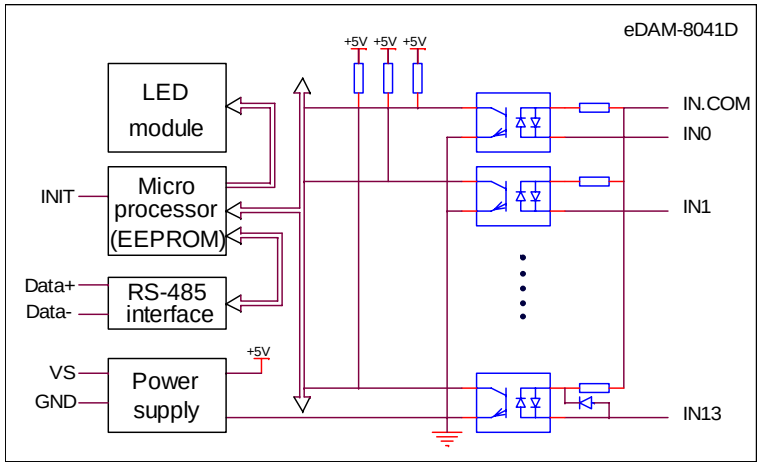
PhotoMos Output module		
Module	Output ch.	Output type
8066(D)	7+1	8 isolated PhotoMos Output channels with normal open

2.4 Block diagram of modules

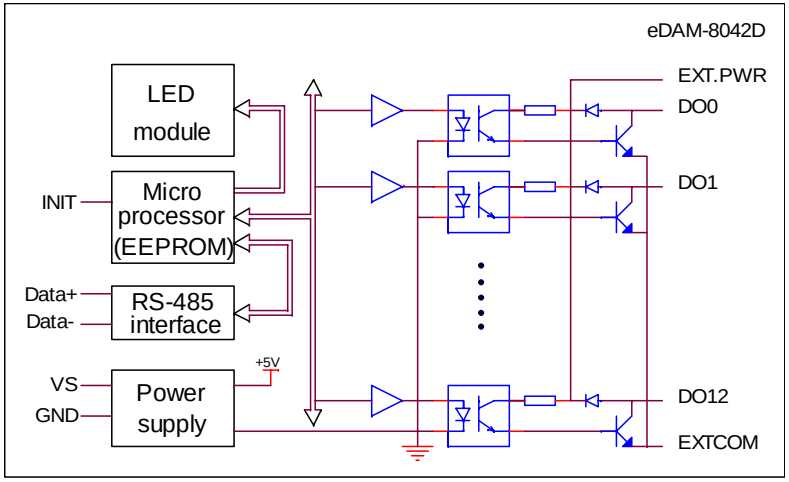
2.4.1 EDAM-8520B



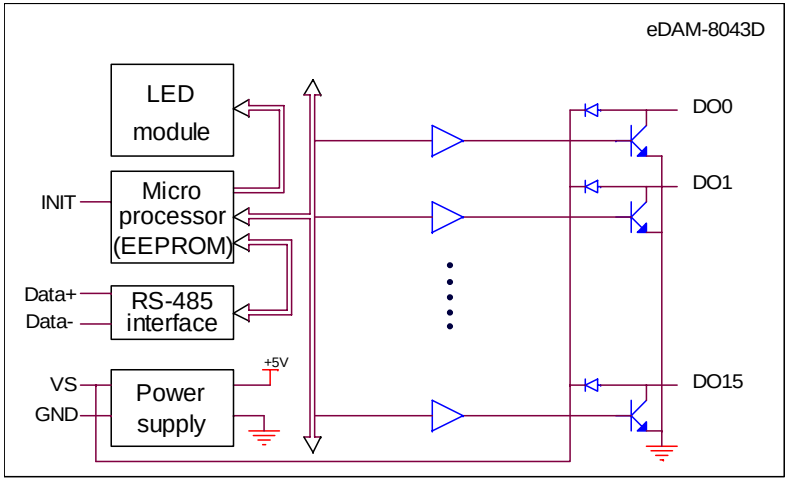
2.4.2 EDAM-8041D



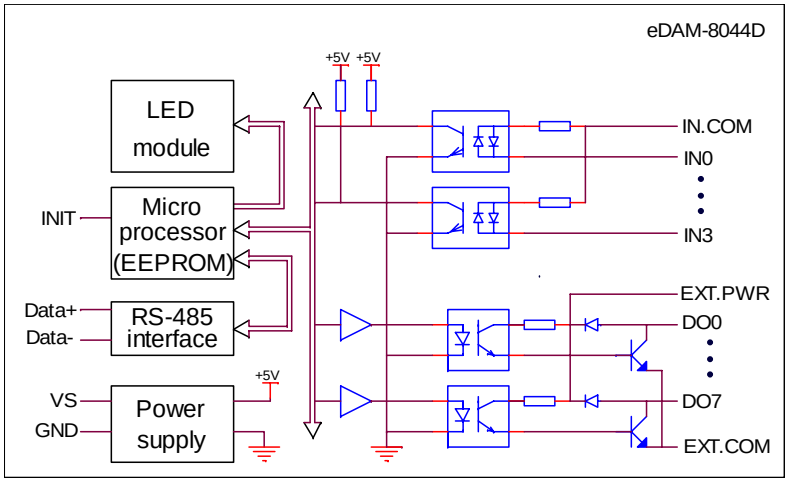
2.4.3 EDAM-8042D



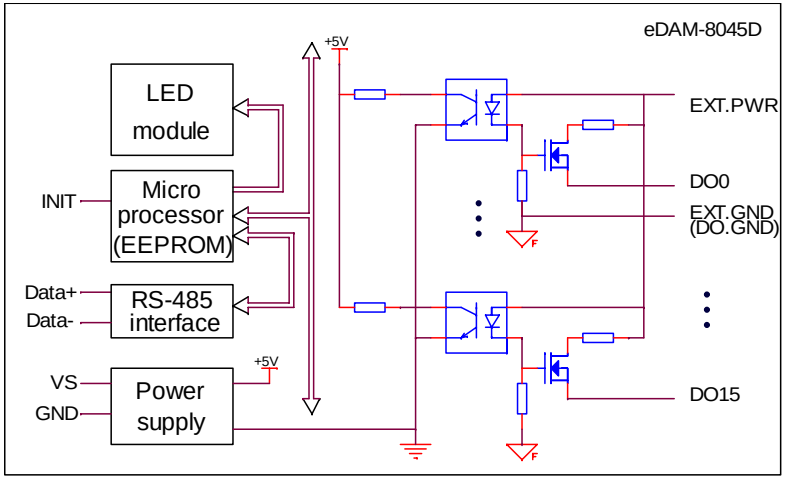
2.4.4 EDAM-8043D



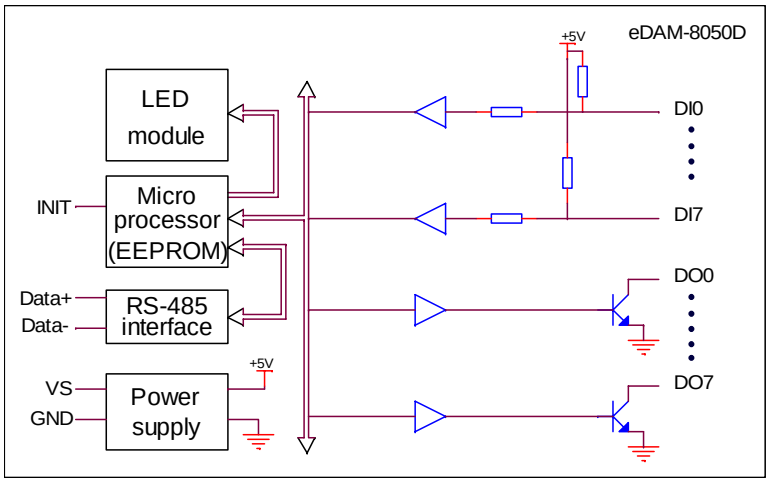
2.4.5 EDAM-8044D



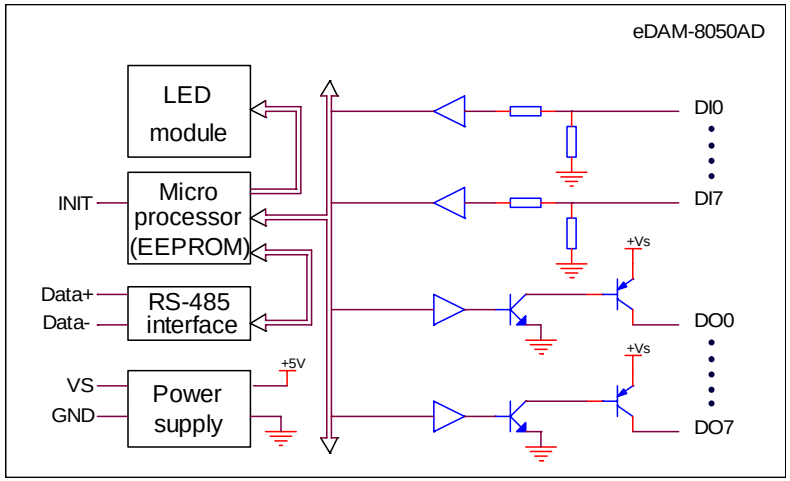
2.4.6 EDAM-8045D



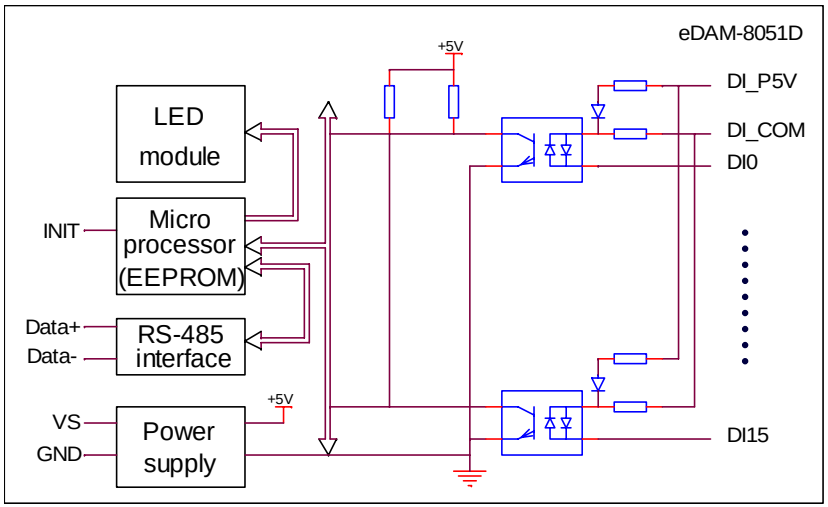
2.4.7 EDAM-8050D



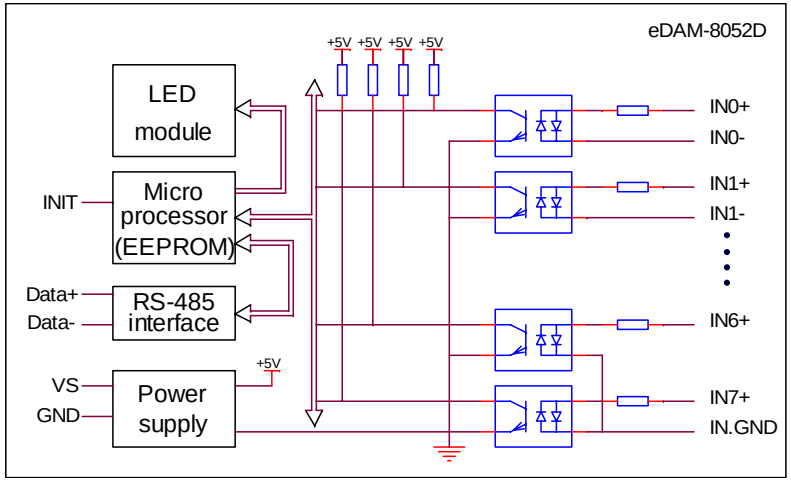
2.4.8 EDAM-8050AD



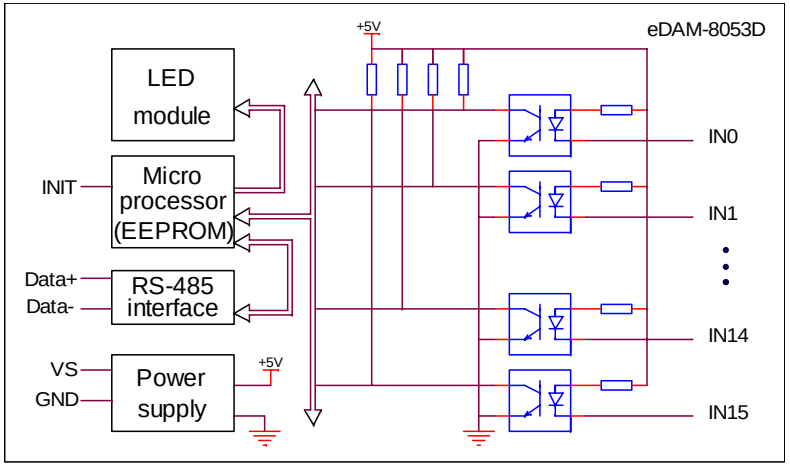
2.4.9 EDAM-8051D



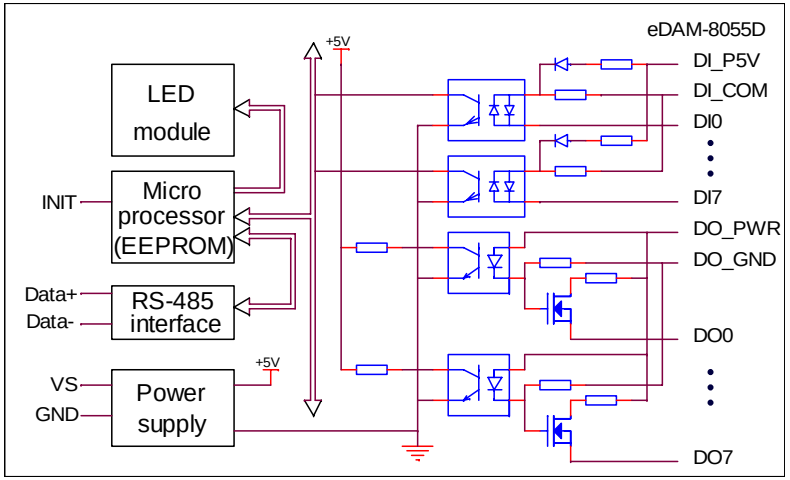
2.4.10 EDAM-8052D



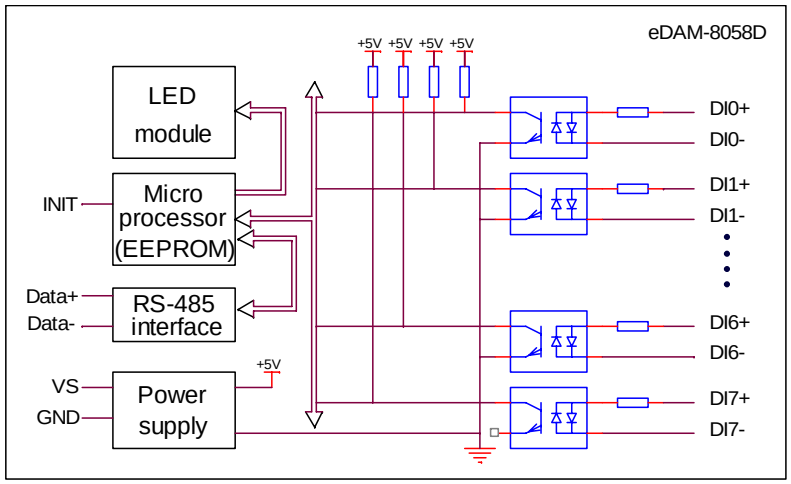
2.4.11 EDAM-8053D



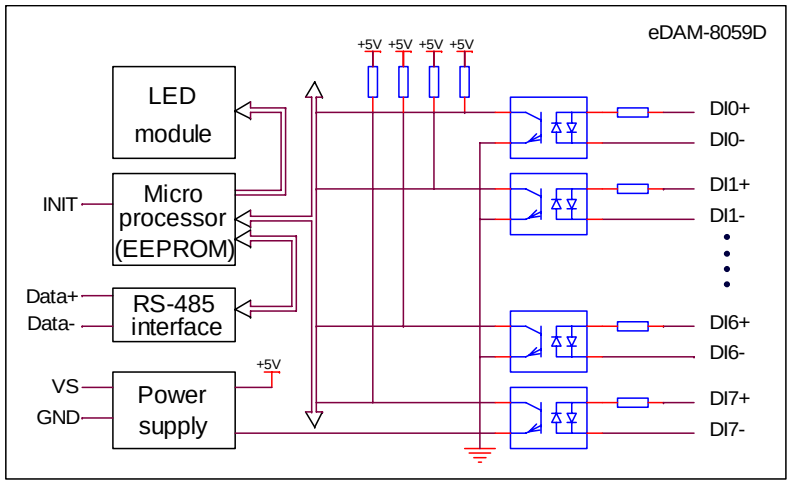
2.4.12 EDAM-8055D



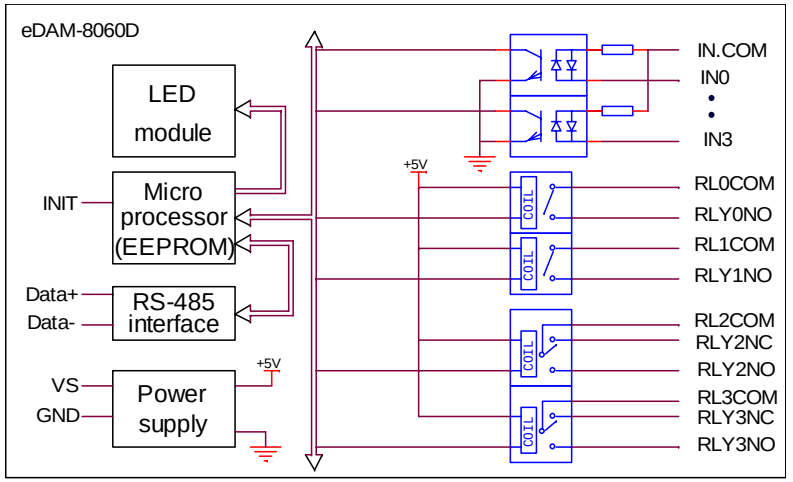
2.4.13 EDAM-8058D



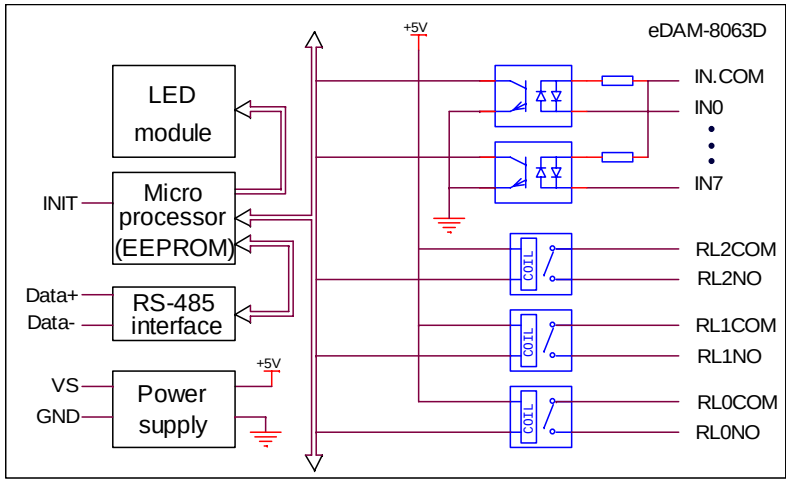
2.4.14 EDAM-8059D



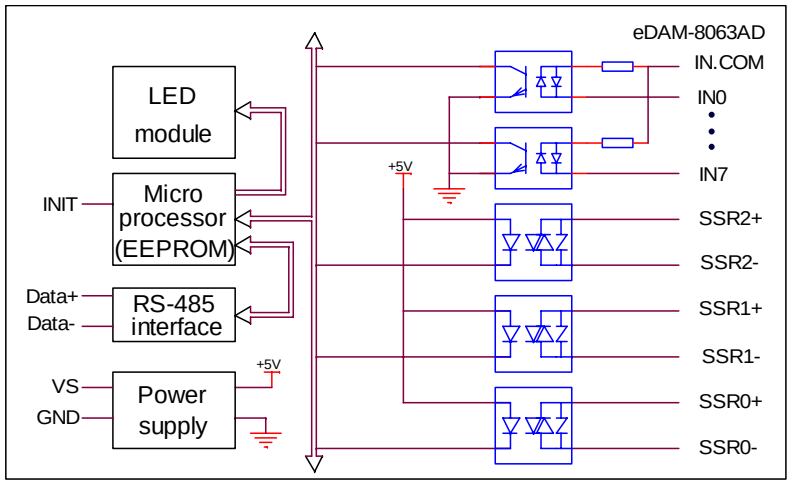
2.4.15 EDAM-8060D



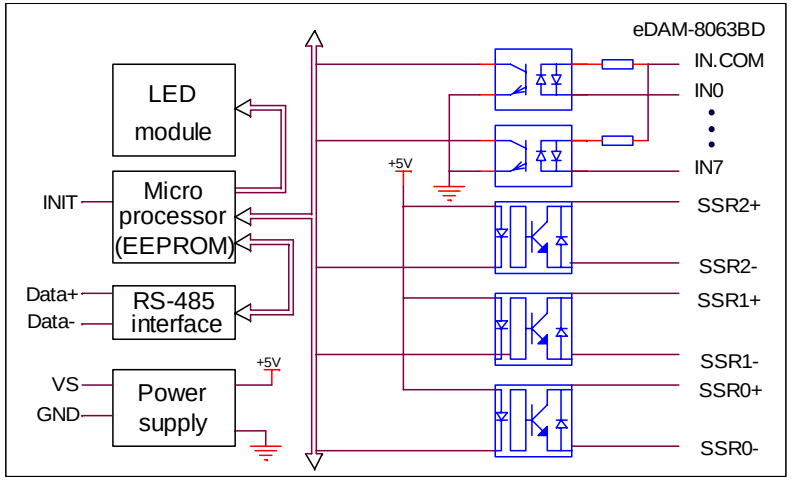
2.4.16 EDAM-8063D



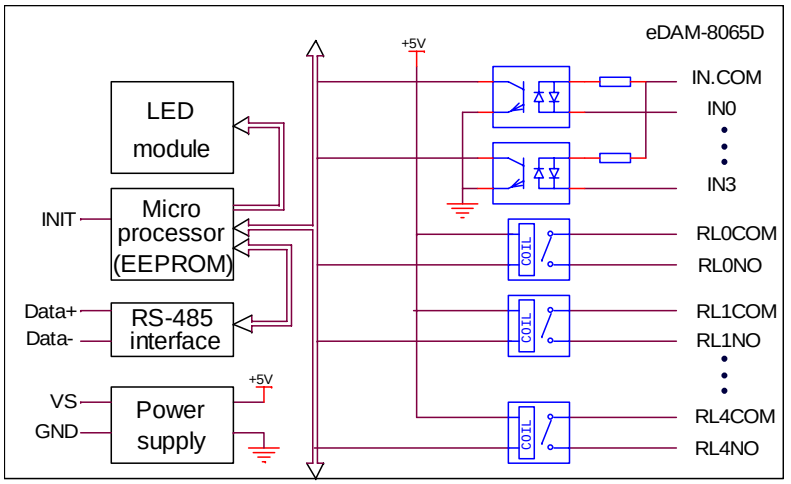
2.4.17 EDAM-8063AD



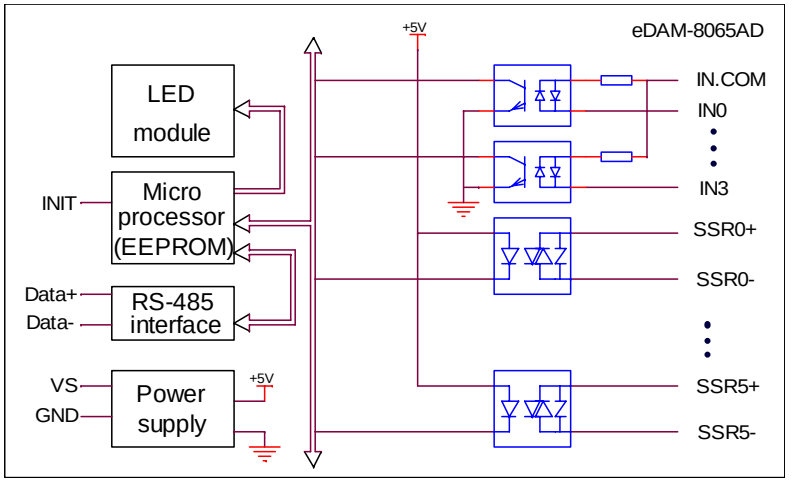
2.4.18 EDAM-8063BD



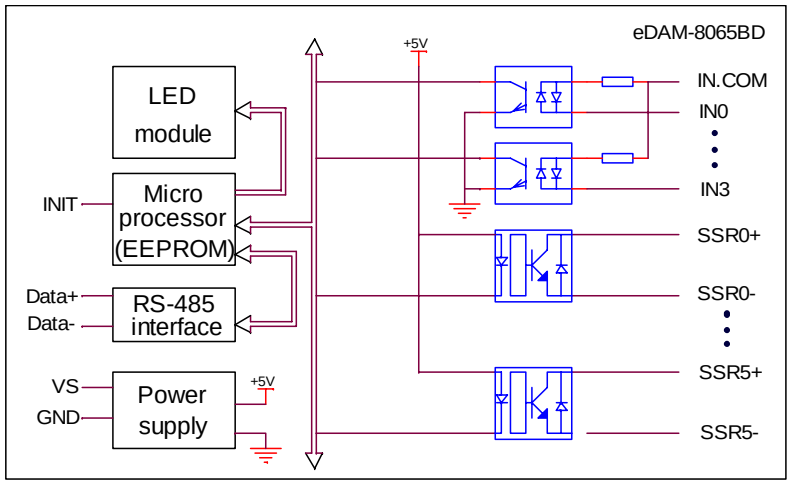
2.4.19 EDAM-8065D



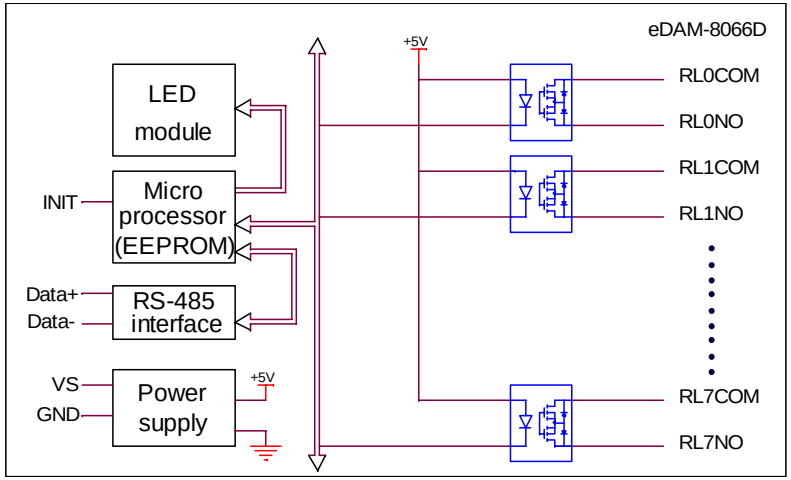
2.4.20 EDAM-8065AD



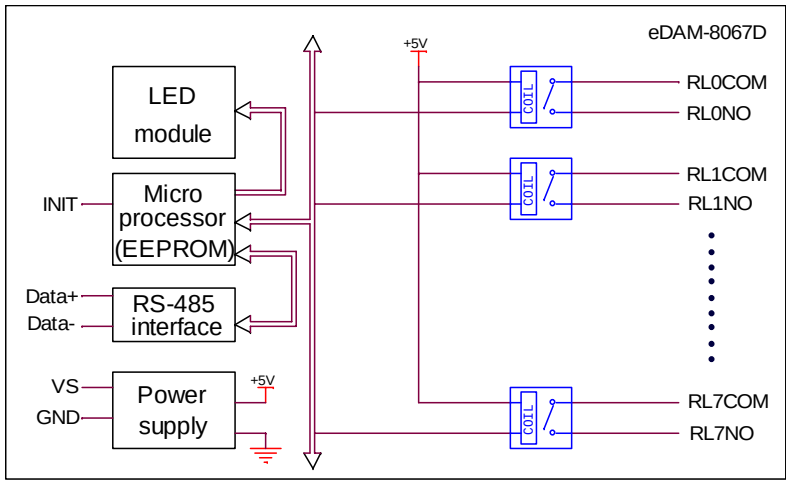
2.4.21 EDAM-8065BD



2.4.22 EDAM-8066D

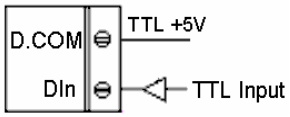
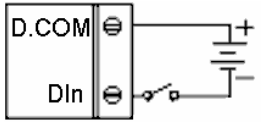
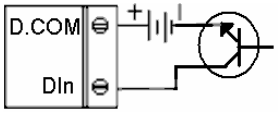
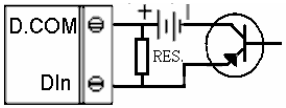


2.4.23 EDAM-8067D

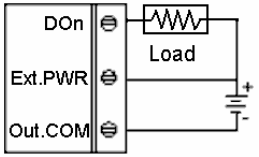
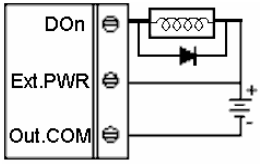


2.5 Wire connection

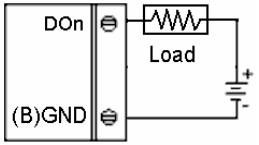
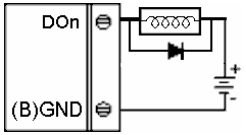
2.5.1 *EDAM-8041D*

TTL/COMS Signal input	Contact Closure input
	
NPN output signal input	PNP output signal input
	

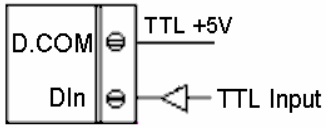
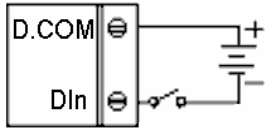
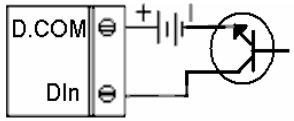
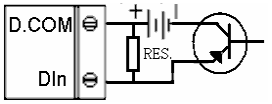
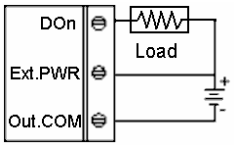
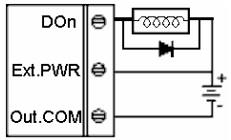
2.5.2 *EDAM-8042D*

Open collector output	
	

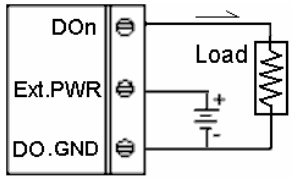
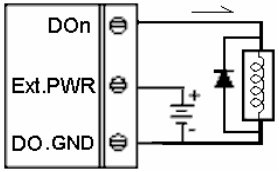
2.5.3 *EDAM-8043D*

Open collector output	
	

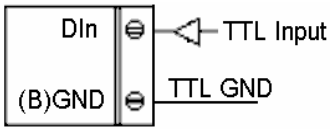
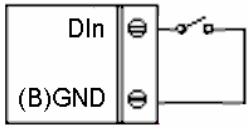
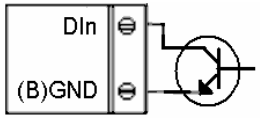
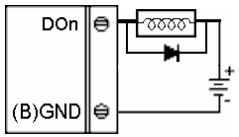
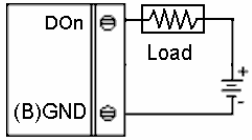
2.5.4 *EDAM-8044D*

TTL/COMS Signal input	Contact Closure input
	
NPN output signal input	PNP output signal input
	
Open collector output	
	

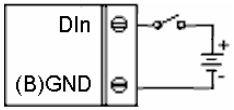
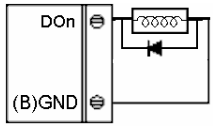
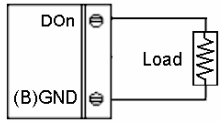
2.5.5 *EDAM-8045D*

Digital output	
	

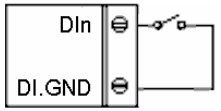
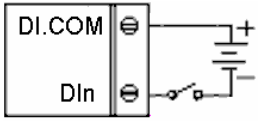
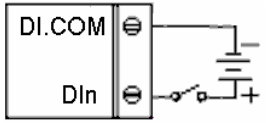
2.5.6 *EDAM-8050D*

TTL/COMS Signal input	Dry Contact Signal input
	
NPN output signal input	
	
Open collector output(NPN)	
	

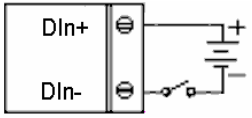
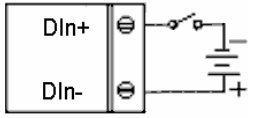
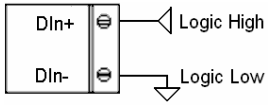
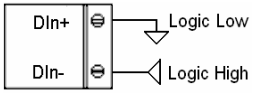
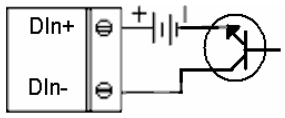
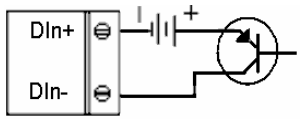
2.5.7 *EDAM-8050AD*

Digital input	
	
Open collector output(PNP)	
	

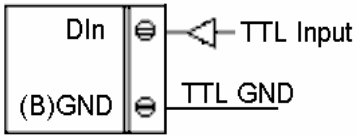
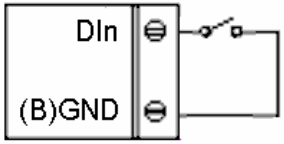
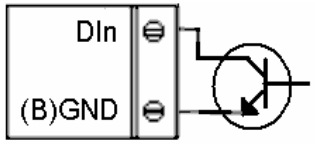
2.5.8 *EDAM-8051D*

Dry Contact input	
	
Web Contact input	
	

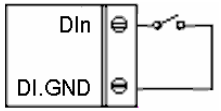
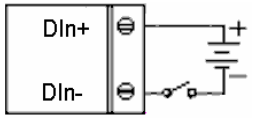
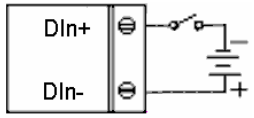
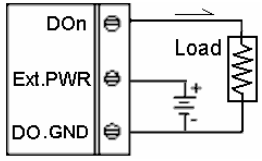
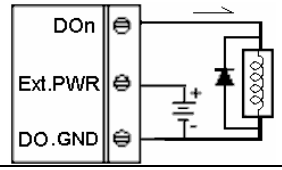
2.5.9 *EDAM-8052D*

Web contact Signal input	
	
TTL/CMOS	
	
NPN output signal input	PNP output signal input
	

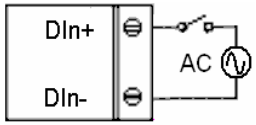
2.5.10 *EDAM-8053D*

TTL/COMS Signal input	Dry Contact Signal input
	
NPN output signal input	
	

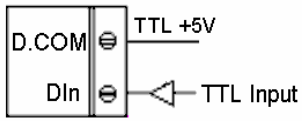
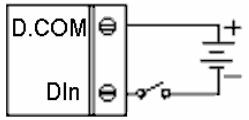
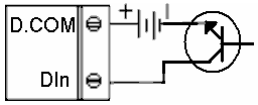
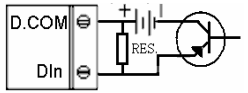
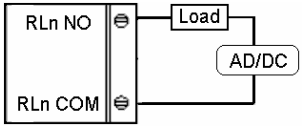
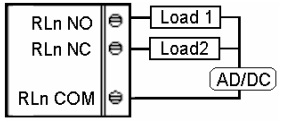
2.5.11 *EDAM-8055D*

Dry Contact Signal input	
	
Web Contact Signal input	
	
Digital output	
	

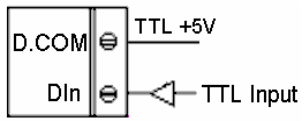
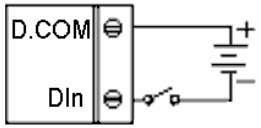
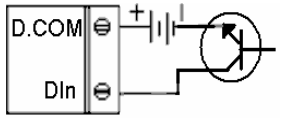
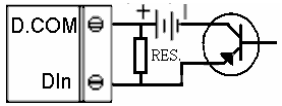
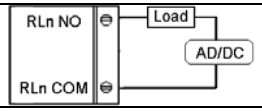
2.5.12 *EDAM-8058D/8059D*

Digital input


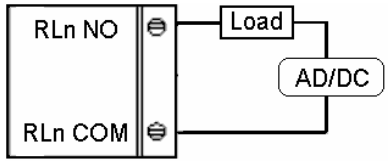
2.5.13 *EDAM-8060D*

TTL/COMS Signal input	Contact Closure input
	
NPN output signal input	PNP output signal input
	
Relay output in RL1 and RL2	Relay output in RL3 and RL4
	

2.5.14 *EDAM-8063D/AD/BD, EDAM-8065D/AD/BD*

TTL/COMS Signal input	Contact Closure input
	
NPN output signal input	PNP output signal input
	
Relay output	
	

2.5.15 *EDAM-8066D, EDAM-8067D*

Relay output


2.6 Specifications

2.6.1 EDAM-8520B module

EDAM-8520B is an isolated RS-232 to RS-422/RS-485 converter, it converts the RS-232 signal to the RS-422/RS-485 signals. The EDAM-8520B equips a "*Auto baud rate detector*" inside, it can detect the baud rate and data format and control the direction of the RS-485 network automatically

Specifications:

- ✧ Input Interface : standard RS-232 9 pin female D-type connector
- ✧ Output Interface : RS-485, differential, 2 half-duplex wires
  RS-422, differential, 4 full-duplex wires
- ✧ Max RS-485 network distance : 4000 ft. (1200m)
- ✧ Speed (bps) : auto switching baud rate
- ✧ Isolation voltage : 3000 Vrms
- ✧ Max loading : 128 EDAMs on a RS-485 network
- ✧ Power supply : +10V to +30V, 0.95 W

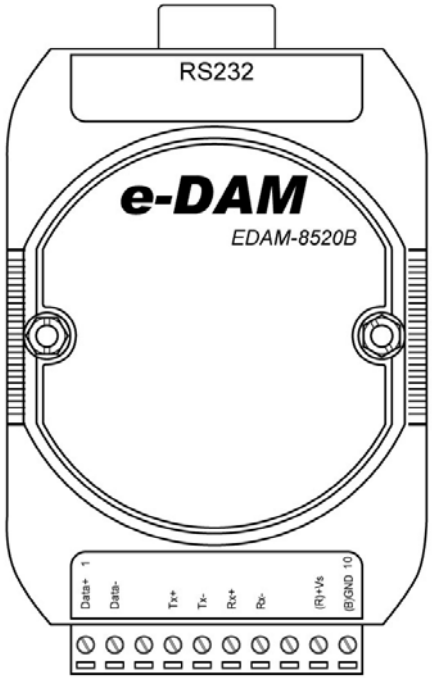
Pin Definitions

RS-232 connector (9-pin D-type female)

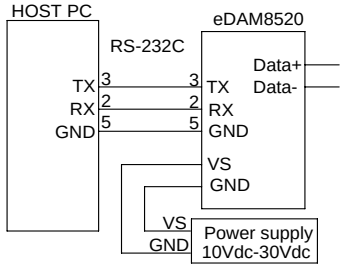
Pin	Name	Description
1	N.C	No used
2	RXD	Receiver
3	TXD	Transmitter
4	N.C	No used
5	GND	Ground
6	N.C	No used
7	N.C	No used
8	N.C	No used
9	N.C	No used

RS-422/485 terminal (10-pin plug-in screw terminal block)

Pin	Name	Description
1	DATA+	RS-485 transmission line, positive
2	DATA-	RS-485 transmission line, negative
3	N.C	No used
4	TX+	RS-422 transmission line, positive
5	TX-	RS-422 transmission line, negative
6	RX+	receiving line, positive
7	RX-	receiving line, negative
8	N.C	No used
9	+Vs	power supply +10V~+30 VDC
10	GND	Power GND



✧ Connection Between Host and EDAM-8520B



✧ Termination Resistor

Termination resistor for DATA+(TX+) & DATA-(TX-) :

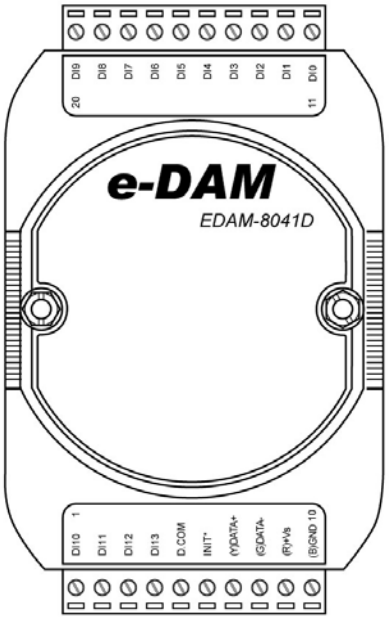
- ✓ If the length RS-485 is about 1.2KM, try 12-ohm first.
- ✓ If the length RS-485 is about 600M, try 220-ohm first.
- ✓ If the length RS-485 is about 300M, try 330-ohm first.

2.6.2 EDAM-8041D module

EDAM-8041D provides 14 isolated digital input channels and all channels are single-ended with common source. The isolation voltage is up to 3750 Vrms. (see sec. 2.4.2 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Channel numbers : 14 isolated single ended with common source
- ✧ Isolation Voltage: 3750Vrms
- ✧ Logical level 0 : +1Vdc Max.
- ✧ Logical level 1: +4.0V ~ +30Vdc
- ✧ Input impedance: 3K ohms
- ✧ LED: 14 digital input status LED
- ✧ Power input : +10V to +30Vdc
- ✧ Power Consumption : 1.2W

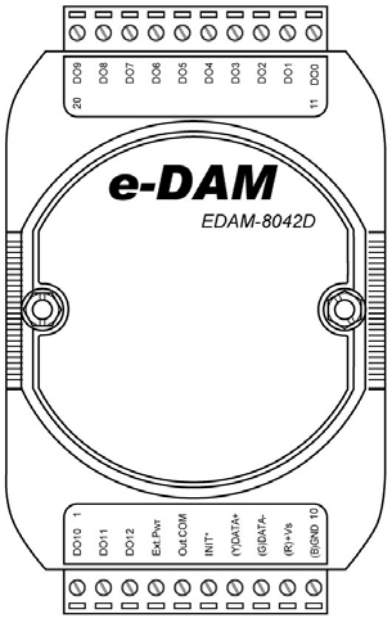


2.6.3 EDAM-8042D module

EDAM-8042D provides 13 isolated digital output(open collector) channels and all channels are single-ended with common power. The isolation voltage is up to 3750 Vrms. (see sec. 2.4.3 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Channel numbers: 13 isolated single end with common power
- ✧ Output characteristic: open collector transistor(NPN)
- ✧ Isolation Voltage: 3750Vrms
- ✧ Max. Load Voltage: +30Vdc
- ✧ Maximum current sink: 100mA
- ✧ LED: 13 digital output status LED
- ✧ Power input : +10V to +30Vdc
- ✧ Power Consumption : 1.6W

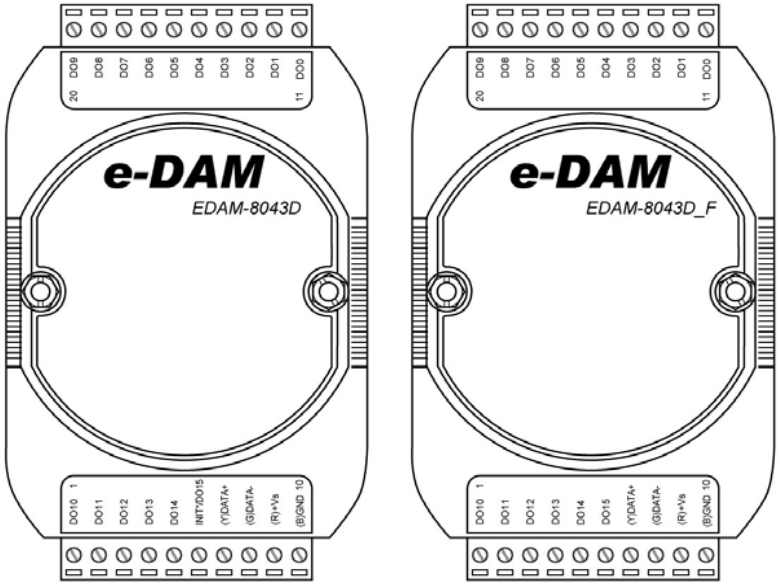


2.6.4 EDAM-8043D module

EDAM-8043D provides 16 non-isolated digital output(open collector) channels and all channels are single-ended with common ground.
(see sec. 2.4.4 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Channel numbers : 16 non-isolated single ended
- ✧ Output characteristic: open collector transistor(NPN)
- ✧ Max Load voltage: 30 Vdc
- ✧ Maximum current sink: 100mA
- ✧ LED: 16 digital output status LED
- ✧ Power input : +10V to +30Vdc
- ✧ Power Consumption : 1.2W



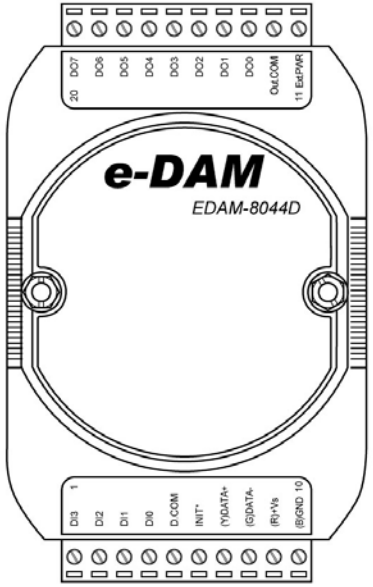
- ◆ For E-8043 Pin-6 are jumper selectable to INIT* or DO15 (Ref. Sec. 3.9)
- ◆ For E-8043F the (INIT*) switch located on the rear side of the module (Ref. Appendix A)

2.6.5 EDAM-8044D module

EDAM-8044D provides 8 isolated digital output(open collector) channels and 4 isolated digital input channels. All output channels are single-ended with common power. (see sec. 2.4.5 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 8 isolated output channels with common power
- ✧ Isolation Voltage: 3750Vrms
- ✧ Output characteristic: open collector transistor(NPN)
- ✧ Max output Load voltage: 30 Vdc
- ✧ Maximum output current sink: 375mA
- ✧ Input channels : 4 isolated input channels with common source
- ✧ Input impedance: 3K ohms
- ✧ Logical level 0 : +1Vdc Max.
- ✧ Logical level 1: +4.0V ~ +30Vdc
- ✧ LED: 12 digital input/output status LED
- ✧ Power input : +10V to +30Vdc
- ✧ Power Consumption : 1.8W

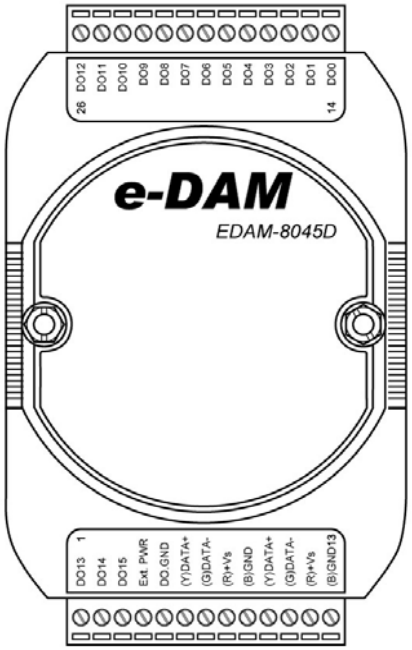


2.6.6 EDAM-8045D module

EDAM-8045D provides 16 isolated digital output(source) channels, All output channels are open source(N-MOSFET). (see. 2.4.6 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 16 isolated output channels (source)
- ✧ Output type: Open source (N-MOSFET)
- ✧ Output load voltage: +10V~+40Vdc
- ✧ Max. load current: 650mA
- ✧ Short-circuit protection: Yes
- ✧ Output isolation Voltage: 3750Vrms
- ✧ LED: 16 digital output status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 3.5W

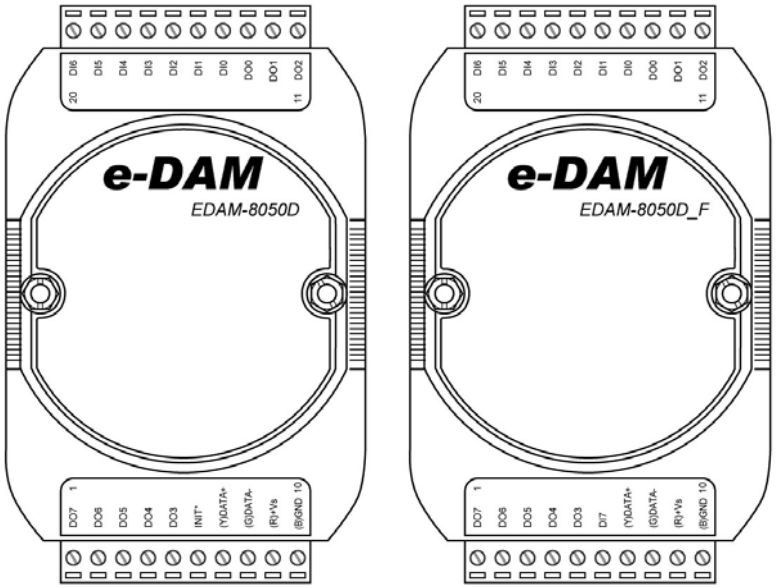


2.6.7 EDAM-8050D module

EDAM-8050D provides 8 non-isolated digital output(open collector/sink) channels and 8(7) non-isolated digital input channels(sink). All input/output channels are single-ended with common ground. (see sec. 2.4.6 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 8 non-isolated output channels(sink)
- ✧ Output characteristic: open collector transistor (NPN)
- ✧ Output Load voltage: +10V ~ +30Vdc
- ✧ Maximum output current sink: 30mA
- ✧ Input channels : 8(7) non-isolated input channels(sink)
- ✧ Logical level 0 : +1Vdc Max.
- ✧ Logical level 1 : +3.5V ~ +30Vdc
- ✧ LED: 16 digital input/output status LED
- ✧ Power input : +10V to +30Vdc
- ✧ Power Consumption : 1.9W



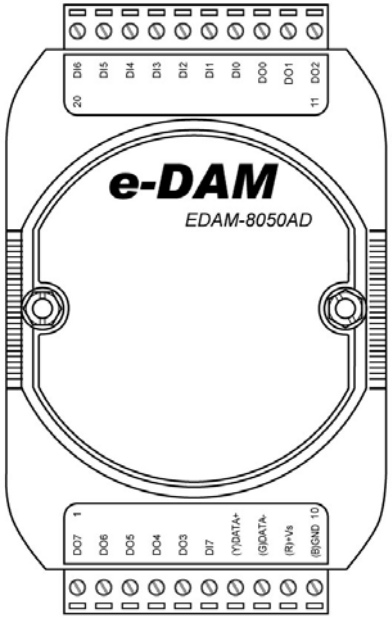
◆ For E-8050F the (INIT*) switch located on the rear side of the module (Ref. Appendix A)

2.6.8 EDAM-8050AD module

EDAM-8050AD provides 8 non-isolated digital output(open collector/source) channels and 8 non-isolated digital input channels. All input/output channels are single-ended with common ground. (see sec. 2.4.8 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 8 non-isolated output channels(source)
- ✧ Output characteristic: open collector transistor (PNP)
- ✧ Output Load voltage: +10V ~ +30Vdc
- ✧ Maximum output current: 50mA
- ✧ Input channels : 8 non-isolated input channels (source)
- ✧ Logical level 0 : +1Vdc Max.
- ✧ Logical level 1 : +4V ~ +30Vdc
- ✧ LED: 16 digital input/output status LED
- ✧ Power input : +10V to +30Vdc
- ✧ Power Consumption : 1.9W

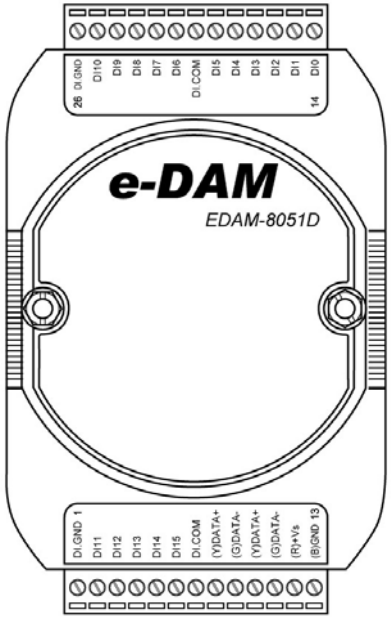


2.6.9 EDAM-8051D module

EDAM-8051D provides 16 isolated digital input channels. All input channels are single ended with common source or common ground. (see sec. 2.4.9 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Input channels : 16 isolated input channels (sink/source).
- ✧ Input type: Isolated single ended with common source or common ground.
- ✧ Logical level 0 : +1Vdc Max.
- ✧ Logical level 1 : +10V ~ +50Vdc
- ✧ Input impedance : 10K ohms
- ✧ Isolation Voltage: 3750Vrms
- ✧ LED: 16 digital input status LED
- ✧ Power input : +10V to +30Vdc
- ✧ Power Consumption : 1.2W

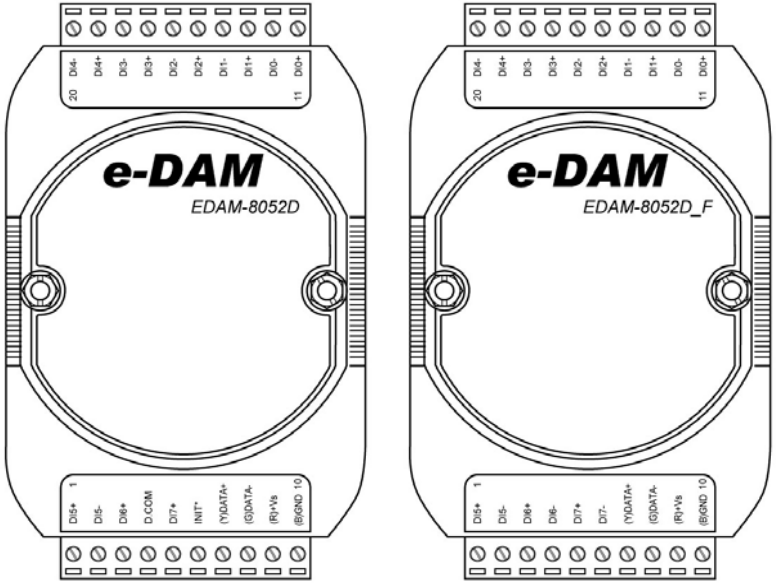


2.6.10 EDAM-8052D module

EDAM-8052D provides 8 isolated differential digital input (**sink/source**) channels. (see sec. 2.4.10 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Input channels: 8 isolated differential input channels **(sink/source)**.
- ✧ Input type: Differential input
- ✧ Isolation Voltage: 3750Vrms
- ✧ Input impedance : 3K ohms
- ✧ Logical level 0 : +1Vdc Max.
- ✧ Logical level 1 : +4.0V ~ +30Vdc
- ✧ LED: 8 digital input status LED
- ✧ Power input : +10V to +30Vdc
- ✧ Power Consumption : 1.2W



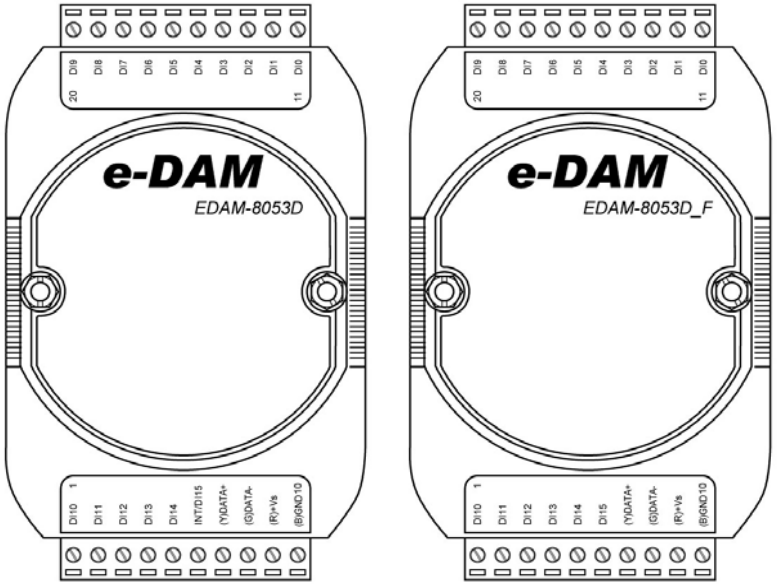
◆ For E-8052F the (INIT*) switch located on the rear side of the module (Ref. Appendix A)

2.6.11 EDAM-8053D module

EDAM-8053D provides 16 non-isolated digital input channels and all channels are single-ended with common ground. (see sec. 2.4.11 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Channel numbers : 16 non-isolated single ended
- ✧ Input impedance : 820 ohms
- ✧ Logical level 0 : +2Vdc Max.
- ✧ Logical level 1: +4.0V ~ +30Vdc
- ✧ LED: 16 digital input status LED
- ✧ Power input : +10V to +30Vdc
- ✧ Power Consumption : 1.7W



◆ For E-8053 Pin-6 are jumper selectable to INIT* or DI15 (Ref. Sec. 3.9)

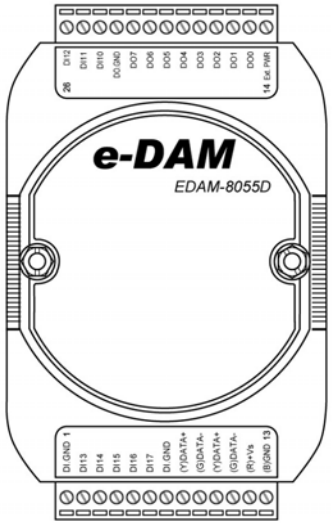
◆ For E-8053F the (INIT*) switch located on the rear side of the module (Ref. Appendix A)

2.6.12 EDAM-8055D module

EDAM-8055D provides 8 isolated digital output(source) channels and 8 isolated digital input(sink/source) channels with common source. All output channels are open source(N-MOSFET). (see sec. 2.4.12 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 8 isolated output channels (source)
- ✧ Output type: Open source(N-MOSFET)
- ✧ Output load voltage: +10V~+40Vdc
- ✧ Max. load current: 650mA
- ✧ Short-circuit protection: Yes
- ✧ Output isolation Voltage: 3750Vrms
- ✧ Input channels: 8 isolated input channels (sink/source)
- ✧ Input type: Isolated single ended with common source or common ground
- ✧ Input impedance: 10K ohms
- ✧ Logical level 0 : +4Vdc Max.
- ✧ Logical level 1: +10V ~ +50Vdc
- ✧ Input isolation Voltage: 3750Vrms
- ✧ LED: 16 digital input/output status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 3.5W

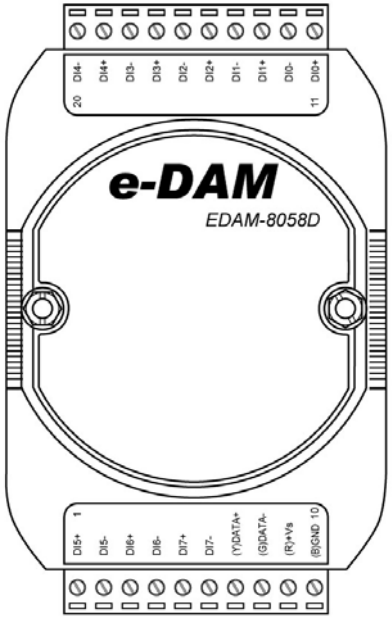


2.6.13 EDAM-8058D module

EDAM-8058D provides 8 isolated differential 250Vac(max) input channels. (see sec. 2.4.13 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Input channels: 8 channels
- ✧ Input type: Differential input
- ✧ Max. input voltage: 250Vac
- ✧ Logical level 0 : <30Vac max.
- ✧ Logical level 1: >80Vac min.
- ✧ Input impedance: 68K ohms
- ✧ Operating AC frequency: 47~400Hz
- ✧ Input isolation Voltage: 3750Vrms
- ✧ LED: 8 digital input status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 1.2W

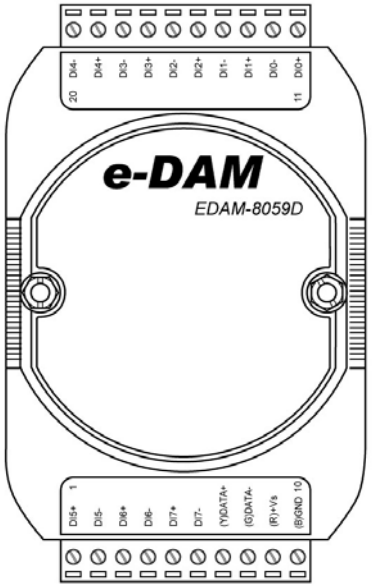


2.6.14 EDAM-8059D module

EDAM-8059D provides 8 isolated differential 80Vac(max) input channels. (see sec. 2.4.14 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Input channels: 8 channels
- ✧ Input type: Differential input
- ✧ Max. input voltage: 80Vac
- ✧ Logical level 0 : <3Vac max.
- ✧ Logical level 1: >10Vac min.
- ✧ Input impedance: 10K ohms
- ✧ Operating AC frequency: 47~400Hz
- ✧ Input isolation Voltage: 3750Vrms
- ✧ LED: 8 digital input status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 1.2W

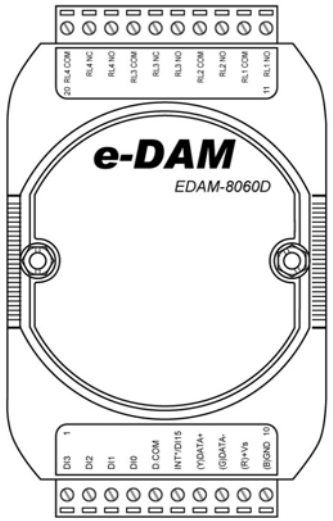


2.6.15 EDAM-8060D module

EDAM-8060D provides 4 isolated digital input channels and 4 relay output channels. all relay output channels are differential with individually common . (see sec. 2.4.15 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels: 4 relay output channels (RL1,RL2: Form A, RL3,RL4 Form C)
- ✧ Relay contact rating : 0.6A/125Vac, 2A/30Vdc
- ✧ Surge strength: 500V
- ✧ Operate Time: 3mS max.
- ✧ Release Time: 2mS max.
- ✧ Min Life: $5*10^5$ ops.
- ✧ Input channels : 4 isolated input channels with common source
- ✧ Isolation Voltage: 3750Vrms.
- ✧ Input impedance: 3K ohms
- ✧ Input logical level 0 : +1V Max.
- ✧ Input logical level 1: +4.0V ~ +30V
- ✧ LED: 8 digital input/output status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 1.8W



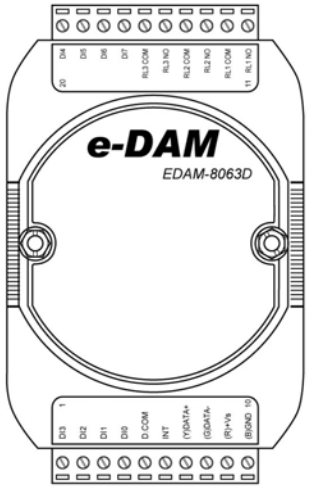
NO: Normal open, NC: Normal Close

2.6.16 EDAM-8063D module

EDAM-8063D provides 8 isolated digital input channels and 3 relay output channels. All input channels are single ended with common source and all relay output channels are differential with individually common .
(see sec. 2.4.16 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 3 relay output channels (Form A)
- ✧ Surge strength: 4000V
- ✧ Relay contact rating : 5A/250Vac, 5A/30Vdc
- ✧ Operate Time: 6mS max.
- ✧ Release Time: 3mS max.
- ✧ Min Life: 10^5 ops.
- ✧ Input channels : 8 isolated input channels with common source
- ✧ Isolation Voltage: 3750Vrms
- ✧ Input impedance: 3K ohms
- ✧ Input logical level 0 : +1V Max.
- ✧ Input logical level 1: +4.0V ~ +30V
- ✧ LED: 11 digital input/output status LED
- ✧ Power input: +10V to +30VDC
- ✧ Power Consumption : 1.8W



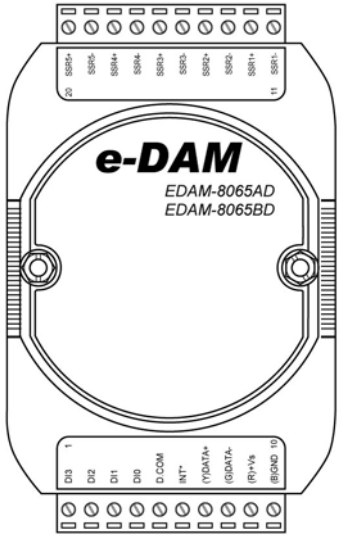
NO: Normal open, NC: Normal Close

2.6.17 EDAM-8063AD module

EDAM-8063AD provides 8 isolated digital input channels and 3 AC-SSR output channels. (see sec. 2.4.17 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 3 AC-SSR output channels with Normal Open
- ✧ Output voltage rating : 24 to 265 Vrms
- ✧ Output current rating : 1.0 Arms,
- ✧ Leakage current: 1.5mArms
- ✧ Min. Operate Time: 1mS,
- ✧ Min. Release Time: 1.5mS
- ✧ Dielectric Strength : 2500Vrms
- ✧ Input channels : 8 isolated input channels with common source
- ✧ Isolation Voltage: 3750Vrms
- ✧ Input impedance: 3K ohms
- ✧ Input logical level 0 : +1V Max.
- ✧ Input logical level 1: +4.0V ~ +30V
- ✧ LED: 11 digital input/output status LED
- ✧ Power input: +10V to +30VDC
- ✧ Power Consumption : 1.8W

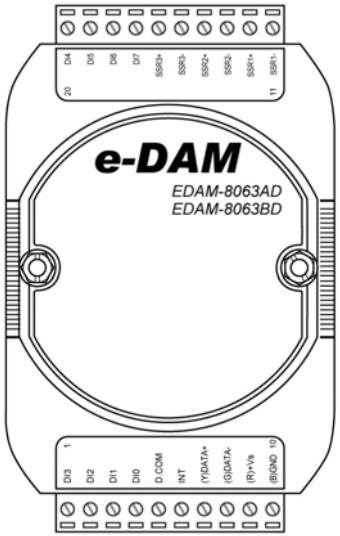


2.6.18 EDAM-8063BD module

EDAM-8063BD provides 8 isolated digital input channels and 3 DC-SSR output channels. All input channels are single ended with common source and all SSR output channels are differential with individually common .
(see sec. 2.4.18 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 3 DC-SSR output channels with Normal Open
- ✧ Output voltage rating : 3 to 30 Vdc
- ✧ Output current rating : 1.0 A,
- ✧ Leakage current: 0.1mA
- ✧ Min. Operate Time: 1mS,
- ✧ Min. Release Time: 1mS
- ✧ Dielectric Strength : 2500Vrms
- ✧ Input channels : 8 isolated input channels with common source
- ✧ Isolation Voltage: 3750Vrms
- ✧ Input impedance: 3K ohms
- ✧ Input logical level 0 : +1V Max.,
- ✧ Input logical level 1: +4.0V ~ +30V
- ✧ LED: 11 digital input/output status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 1.8W

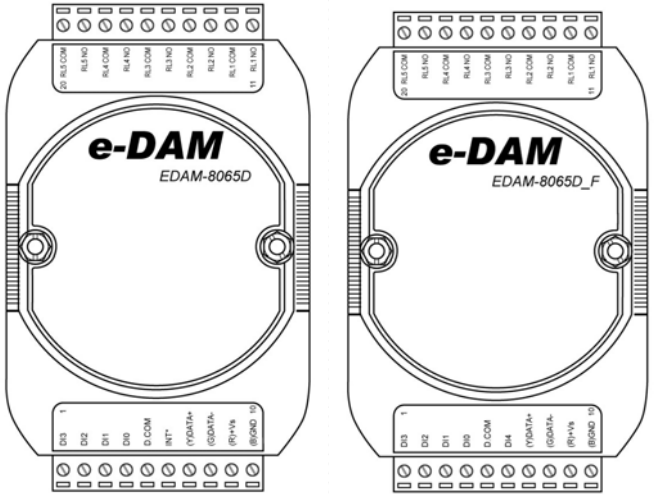


2.6.19 EDAM-8065D module

EDAM-8065D provides 5(4) isolated digital input channels and 5 relay output channels. All input channels are single ended with common source and all relay output channels are differential with individually common .
(see sec. 2.4.19 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 5 relay output channels (Form A)
- ✧ Relay contact rating : 5A/250Vac, 5A/30Vdc
- ✧ Surge strength: 4000V
- ✧ Operate Time: 6mS max.
- ✧ Release Time: 3mS max.
- ✧ Min Life: 10^5 ops
- ✧ Input channels : 5(4) isolated input channels with common source
- ✧ Isolation Voltage: 3750Vrms
- ✧ Input impedance: 3K ohms
- ✧ Input logical level 0: +1V Max.,
- ✧ Input logical level 1: +4.0V ~ +30V
- ✧ LED: 10 digital input/output status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 2.4W



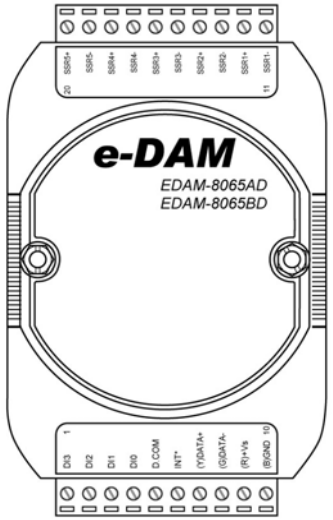
- NO: Normal open, NC: Normal Close
- For 8065F (INIT*) switch located on the rear side of the module(Appendix A)

2.6.20 EDAM-8065AD module

EDAM-8065AD provides 4 isolated digital input channels and 5 AC-SSR output channels. All input channels are single ended with common source.
(see sec. 2.4.20 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 5 AC-SSR output channels with Normal Open
- ✧ Output voltage rating : 24 to 265 Vrms
- ✧ Dielectric Strength : 2500Vrms
- ✧ Output current rating : 1.0 Arms,
- ✧ Leakage current: 1.5mArms
- ✧ Min. Operate Time: 1mS max.
- ✧ Min. Release Time: 1.5mS max.
- ✧ Input channels : 4 isolated input channels with common source
- ✧ Isolation Voltage: 3750Vrms
- ✧ Input impedance: 3K ohms
- ✧ Input logical level 0 : +1V Max.,
- ✧ Input logical level 1: +4.0V ~ +30V
- ✧ LED: 9 digital input/output status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 2.4W

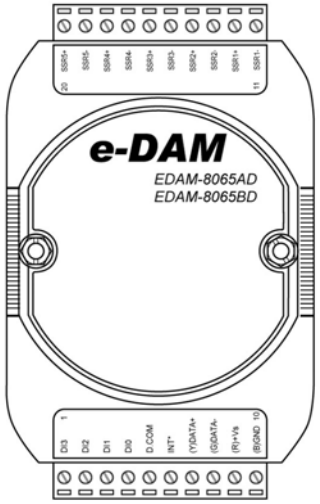


2.6.21 EDAM-8065BD module

EDAM-8065BD provides 4 isolated digital input channels and 5 DC-SSR output channels. All input channels are single ended with common source.
(see sec. 2.4.21 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 5 DC-SSR output channels with Normal Open
- ✧ Output voltage rating : 3 to 30 Vdc
- ✧ Output current rating : 1.0A,
- ✧ Leakage current: 0.1mA
- ✧ Min. Operate Time: 1mS max.
- ✧ Min. Release Time: 1mS max.
- ✧ Dielectric Strength : 2500Vrms
- ✧ Input channels : 4 isolated input channels with common source
- ✧ Isolation Voltage: 3750Vrms
- ✧ Input impedance: 3K ohms
- ✧ Input logical level 0 : +1V Max.,
- ✧ Input logical level 1: +4.0V ~ +30V
- ✧ LED: 9 digital input/output status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 2.4W



2.6.22 EDAM-8066D module

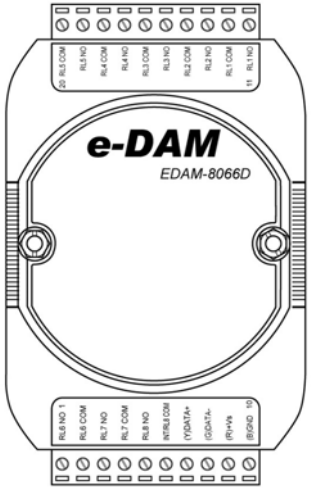
EDAM-8066D provides 8-ch isolated PhotoMOS relays output for control of low-level analog signals without distortion. (see sec. 2.4.22 Block diagram)

Typical applications:

- High-speed inspection machines
- Telephone equipment
- Data communication equipment

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 8 PhotoMOS output CH. with Normal Open
- ✧ Turn-On time: 0.7ms.,
- ✧ Turn-Off time: 0.05ms.
- ✧ Out RES.: 23 ohms.,
- ✧ I/O CAP. : 0.8pf (f=1Mhz)
- ✧ Output rating : 0~350 VAC max (peak), 0~0.13 A max (peak)
- ✧ Isolation Voltage: 5000VAC
- ✧ LED: 8 digital output status LED
- ✧ Power input : +10V to +30VDC,
- ✧ Power Consumption : 1.4W



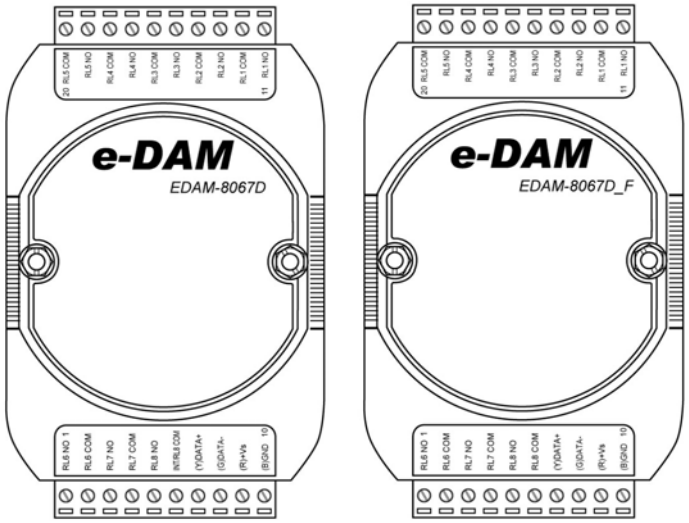
- Signal assignment of Pin-6 is jumper selectable by setting JP1 in module (Ref. Sec. 3.9)

2.6.23 EDAM-8067D module

EDAM-8067D provides 8 channel relay outputs. all output channels are differential with individually common. (NO: Normal open, NC: Normal Close) (see sec. 2.4.23 Block diagram)

Specifications

- ✧ Interface : RS-485, 2 wires
- ✧ Speed : 1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K,115.2K
- ✧ Output channels : 8 relay output channels with Normal Open(Form A)
- ✧ Relay contact rating : 0.5A/120Vac, 1.0A/24Vdc
- ✧ Surge strength: 1500V
- ✧ Operate Time: 5mS max.
- ✧ Release Time: 2mS max.
- ✧ Min Life: 10^5 ops.
- ✧ LED: 8 relay output status LED
- ✧ Power input : +10V to +30VDC
- ✧ Power Consumption : 1.4W



- For E-8067 signal assignment of Pin-6 is jumper selectable by setting JP1 in module (Ref. Sec. 3.9)
- For E-8067_F the (INIT*) switch located on the rear side of the module (Ref. Appendix A)

Chapter 3 Installation

This chapter provides guidelines to what is needed to set up and install an EDAM network. A quick hookup scheme is provided that lets you configure modules before they are installed in a network.

To help you to connect EDAM modules with sensor inputs, several wiring examples are provided. Finally, you will find at the end of this chapter a programming example using the EDAM command set.

Be sure to carefully plan the layout and configuration of your network before you start. Guidelines regarding layout are given in Appendix A:.

3.1 Set up an EDAM network

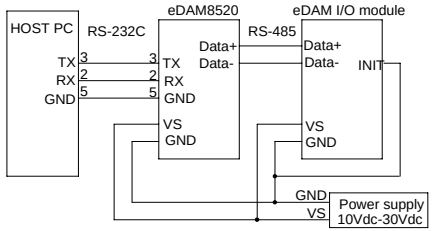
The following list gives an overview of what is needed to setup, install and configure an EDAM environment.

- EDAM modules
- A host computer that can output characters with an RS-232C or RS-485 port.
- Power supply for the EDAM modules (+10 to +30 VDC)
- EDAM Series Utility software
- EDAM Isolated RS-232/RS-485 Converter (optional)
- RS-232/RS-485 EDAM Repeater (optional)

3.2 Host computer

Any computer or terminal that can output characters over either RS-232 or RS-485 can be connected as the host computer. When only RS-232 is available, an EDAM-8520B module (RS-232/RS-485 converter) is required to transform the host signals to the correct RS-485 protocol. The converter also provides opto-isolation and transformer-based isolation to protect your equipment.

For the ease of use in industrial environments the EDAM modules are designed to accept industry standard +24VDC unregulated power. Operation is guaranteed when using any power supply between +10 and +30VDC. Power ripples must be limited to 5 V peak to peak while the voltage in all cases must be maintained between +10 and +30 VDC . All power supply specifications are referenced at module connector. When modules are powered remotely, the effects of line voltage drops must be considered.

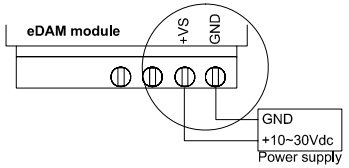


3.3 Power supply

All modules use on-board switching regulators to sustain good efficiency over the +10 ~ +30VDC input range, therefore we can assume that the actual current draw is inversely proportional to the line voltage. The following example shows how to calculate the required current that a power supply should be able to provide.

We advise the following standard colors (as indicated on the modules) for each power line:

+Vs	(R)	Red
GND	(B)	Black

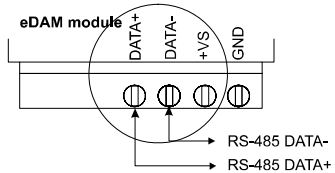


3.4 Communication Wiring

We recommend that shielded-twisted-pair cables that comply with the EIA RS-485 standard be used with the EDAM network to reduce interference.

We advise the following standard colors (as indicated on the modules) for each power line:

DATA+	(Y)	Yellow
DATA-	(G)	Green



3.5 EDAM Utility Software

A menu-driven utility program for DOS or Windows is provided for EDAM module configuration, monitoring and calibration. It also includes a terminal emulation program that lets you easily communicate through the EDAM command set

3.6 EDAM Isolated RS-232/RS485 Converter (optional)

When the host computer or terminal has only a RS-232 port, an EDAM-8520B Isolated RS-232/RS-485/422 converter connected to the host's RS-232 port is required.

This module equips a "Auto baud rate detector" inside, therefore it can detect the baud rate and data format automatically and control the direction of RS-485 precisely.

3.7 EDAM Repeater (optional): EDAM-8510

When communication lines exceed 4000 ft (1200 meter) or more than 32 EDAM modules are connected, a repeater should be implemented. In a network, up to eight Repeater modules can be connected allowing connection up to 255 EDAM modules. As with the Converter module, the Repeater module is not addressable by the host and the baud rate must be reset by changing the switch inside the module. The factory default setting is 9600 baud.

3.8 Initializing a Brand-New Module

All EDAM modules in a RS-485 network must have an *unique* address ID. Therefore, to configure the brand-new EDAM before using is necessary.

Factory default settings:

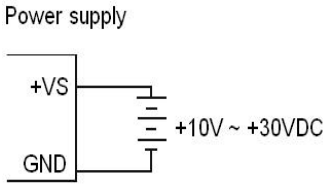
- Address ID is 01
- Baud rate is 9600 bps (N,8,1)
- Check-sum disable
- EDAM-ASCII protocol

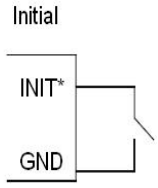
INIT* State settings:

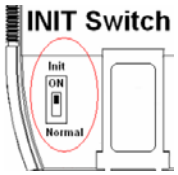
The EDAM I/O modules must be set at ***INIT* State*** when you want to change the default settings, such as the *ID address, baud rate, edam-ascii protocol, check-sum status* etc. All EDAM I/O modules have an special pin labeled as **INIT*** ***(ref. Appendix A)***. The module will be in *Default State* if the **INIT*** pin is shorted to ground(or INIT switch ON) when power ON. Under this state, the default configuration is set as following :

- Address ID is 00
- Baud rate is 9600 bps (N,8,1)
- Check-sum disable
- EDAM-ASCII protocol

Therefore, the communication between host and the module will can be easily set as the same configuration, the initialization of a module will be possible no matter what configuration is set under operating state.

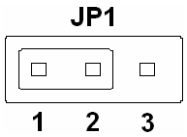




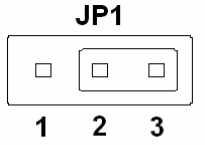


3.9 Jumper setting

For EDAM-8043D, EDAM-8053D, the pin-6 is used for both DO15 (DI15) and INIT* (default).
For EDAM-8066D, EDAM-8067D, the pin-6 is used for both RL8_COM and INIT* (default)
When you want to use pin 6 of EDAM-8043D/8053D as DO15(DI15) or pin 6 of EDAM-8066D/8067D as RL8_COM, you should open the module case to set the JP1.
Note: To access the jumper, the cover must be opened.



Pin-6 = RL8 COM



Pin-6 = INIT*
(factory default)

3.10 Initialization Procedure

1. Connect a brand new EDAM module with the RS-485. Set the module in *Default State* by shorting the **INIT*** pin to GND.(see 3.8)
2. Power on the power supply for EDAM modules.
3. Use the EDAM utility to configure the address ID, baud rate, check-sum status and command sets of the module.

3.11 Changing the protocol from EDAM ASCII to Modbus-RTU

Changing the protocol from EDAM ASCII to Modbus Some EDAM-8000 modules support both EDAM ASCII and Modbus protocols, and the factory default setting of these modules is EDAM ASCII protocol. If you would like to configure the modules to Modbus protocol, please refer to Appendix G which describes how to change the protocol in EDAM utility.

To switch to the Modbus RTU protocol: (see Appendix Appendix G)

1. Sends the $AAPN command and set N to a value of 1.
   Note: It is necessary to short the pin **INIT*** to ground. (see 3.8)
2. After a power-on reset, the communication protocol will be changed to the Modbus-RTU protocol.

To switch to the EDAM-ASCII format protocol:

1. Uses address 00257 of Modbus function and set to a value of 0.
2. After a power-on reset, the communication protocol will be changed to EDAM-ASCII format protocol.

3.12 Install a New EDAM to a Existing Network

1. Equipments for Install a New Module
2. A existing EDAM network
3. New EDAM modules.
4. Power supply (+10 to +30 VDC)

Installing Procedures

1. Configure the new EDAM module according to the initialization procedure in Appendix A.
2. The baud rate and check-sum status of the new module must be identity with the existing RS-485 network. The address ID must not be conflict with other EDAM modules on the network.
3. Power off the EDAM power supply of the existing RS-485 network.
4. Wire the power lines for the new EDAM with the existing network. Be careful about the signal polarity as wiring.
5. Wire the RS-485 data lines for the new EDAM with the existing network. Be careful about the signal polarity as wiring.
6. Wire to the input or output devices.
7. Power on the EDAM local power supply.
8. Use the EDAM utility to check entire network.

3.13 Configuration Table of EDAM DIO module (see 4.5)

3.13.1 Baud Rate settings (CC)

Code	03	04	05	06	07	08	09	0A
Baud rate	1200	2400	4800	9600	19200	38400	57600	115200

Note: The data bits are fixed at one start bit, eight data bits, no parity and one stop bit.

3.13.2 Data format settings (FF)

bit	7	6	5	4	3	2	1	0

Bit7: input counter update direction: (see "#AAN" in 4.5.27)
0=Falling edge (default)
1=rising edge

Bit6: checksum:
1=Enable
0=disable (default)

Bit5~bit0: reserved must be 0

Note: It's needed to short the **INIT*** pin to ground while changing baud rate and/or enable/disable checksum.

3.14 Digital Input/Output Data Format Table

The data format of the response of the $AA4, $AA6 and $AALS commands is:
(the First Data)(the Second Data)00.

The data format of the response of the @AA command is:
(the First Data)(the Second Data).

Note: both the First Data and the Second Data are in two hexadecimal digits format.

Module	The First data		The Second data	
EDAM-8041D	DI8~DI13	00~3F	DI0~DI7	00~FF
EDAM-8042D	DO8~DO12	00~1F	DO0~DO7	00~FF
EDAM-8043D	DO8~DO15	00~FF	DO0~DO7	00~FF
EDAM-8044D	DO0~DO7	00~FF	DI0~DI3	00~0F
EDAM-8045D	DO8~DO15	00~FF	DO0~DO7	00~FF
EDAM-8050AD	DO0~DO7	00~FF	DI0~DI7	00~FF
EDAM-8051D	DI8~DI15	00~FF	DI0~DI7	00~FF
EDAM-8052D	DI0~DI7	00~FF		00
EDAM-8053D	DI8~DI15	00~FF	DI0~DI7	00~FF
EDAM-8055D	DO0~DO7	00~FF	DI0~DI7	00~FF
EDAM-8058D	DI0~DI7	00~FF		00
EDAM-8059D	DI0~DI7	00~FF		00
EDAM-8060D	RL1~RL4	00~0F	DI0~DI3	00~0F
EDAM-8063D	RL1~RL3	00~07	DI0~DI7	00~FF
EDAM-8063AD	AC SSR1~ SSR3	00~07	DI0~DI7	00~FF
EDAM-8063BD	DC SSR1~ SSR3	00~07	DI0~DI7	00~FF
EDAM-8065D	RL1~RL5	00~1F	DI0~DI4	00~1F
EDAM-8065AD	AC SSR1~ SSR5	00~1F	DI0~DI3	00~0F
EDAM-8065BD	DC SSR1~ SSR5	00~1F	DI0~DI3	00~0F
EDAM-8066D	RL1~ RL8	00~FF		00
EDAM-8067D	RL1~ RL8	00~FF		00

3.15 DIO Active States

The DIO read value of the EDAM-8000 is as follows:

Module	DIO	Inactive	Active
EDAM-8041D	14 DI	ON	OFF
EDAM-8042D	13 DO	OFF	ON
EDAM-8043D	16 DO	OFF	ON
EDAM-8044D	8 DO	OFF	ON
	4 DI	ON	OFF
EDAM-8045D	16 DO	OFF	ON
EDAM-8050D	8 DO	OFF	ON
	8 DI	ON	OFF
EDAM-8050AD	8 DO	OFF	ON
	8 DI	OFF	ON
EDAM-8051D	16 DI	OFF	ON
EDAM-8052D	8 DI	OFF	ON
EDAM-8053D	16 DI	ON	OFF
EDAM-8055D	8 DO	OFF	ON
	8 DI	OFF	ON
EDAM-8058D	8 DI	OFF	ON
EDAM-8059D	8 DI	OFF	ON
EDAM-8060D	4 DO	OFF	ON
	4 DI	ON	OFF
EDAM-8063D/AD/BD	3 DO	OFF	ON
	8 DI	ON	OFF
EDAM-8065D	5 DO	OFF	ON
	5 DI	ON	OFF
EDAM-8065AD/BD	5 DO	OFF	ON
	4 DI	ON	OFF
EDAM-8066D	8 DO	OFF	ON
EDAM-8067D	8 DO	OFF	ON

ON means the DIO read value is 1.
OFF means the DIO read value is 0.

Chapter 4 EDAM-ASCII protocol Command Sets

4.1 Introduction

The EDAM-ASCII command is composed by numbers of characteristics, including the leading code, address ID, the variables, the optional check-sum byte, and a carriage return to indicate the end of a command.
The host computer can only command only one EDAM module except those synchronized commands with wildcard address commands "#**" and "~**". The EDAM may or may not give response to the command. The host should check the response to handshake with the modules.

4.2 Format of EDAM ASCII Commands

Syntax: (Leading code)(Addr)(Command)[Data] <Cksum><CR>

Every command begins with a delimiter character. There are five valid characters: a dollar sign $, a pound sign #, a percentage sign %, a wave sign ~ and an at sign @.
The delimiter character is followed by a two-character address (hexadecimal) that specifies the target module. The actual two character command follows the address. Depending on the command, an optional data segment follows the command string. An optional two character checksum may be appended to the total string. Every commands is terminated by a carriage return (cr).
Conventions

Leading Code	The first characteristic of the EDAM command, such as %, $, #, ~, @, ...etc(1- character)
Addr	Module's address ID, the value is in the range of 00 – FF (Hex) 2- character
Command	Command codes or value of variables
Data	Data needed by some output command
Checksum	Checksum in brackets indicate optional parameter, only checksum is enable then this field is required (2- character)
<CR>	carriage return(0x0D)

Note:
all commands should be issued in **UPPERCASE** characters !
2. There is no spacing between characters.

Calculate Checksum:

1. Calculate ASCII sum of all characters of command (or response) string except the character return(cr)
2. Mask the sum of string with 0ffh
   [Checksum] = {(Leading code)+(addr)+(command)+[data]} MOD 0x100

Example:

Command string : $012(cr)
Sum of string = '$'+'0'+'1'+'2'=24h+30h+31h+32h=B7h
The checksum is B7h, and [CHK] = "B7"
Command string with checksum = $012B7(cr)

Response string : !01400600(cr)
Sum of string = '!'+'0'+'1'+'4'+'0'+'0'+'6'+'0'+'0'
= 21h+30h+31h+34h+30h+30h+36h+30h+30h=1ACh
The checksum is ACh, and [CHK] = "AC"
Response string with checksum = !01400600AC(cr)

4.3 Response of Commands

The response message depends on EDAM command. The response is also composed with several characteristics, including leading code, variables, and carriage return for ending. There are two kinds of leading code for response message, "!" or ">" means valid command and "?" means invalid. By checking the response message, user can monitor the command is valid or invalid. But under the following conditions, there will have no response message.

- The specified address ID is not exist.
- Syntax error.
- Communication error
- Some special commands does not have response.

4.4 Table of Command sets

■ General Commands

Syntax	Description	Modules	Page
%AANNTTCCFF	Sets the module configuration	All EDAM modules	68
$AA2	Reads the module configuration	All EDAM modules	71
~AAI	Soft INIT command	All EDAM modules	72
~AATnn	Sets the soft INIT timeout value	All EDAM modules	73
$AA5	Reads the Reset Status of a module	All EDAM modules	75
$AAF	Read the firmware version of a module	All EDAM modules	76
$AAM	Reads the module name	All EDAM modules	77
$AAO(data)	Sets the module name	All EDAM modules	78
$AAPN	Sets the communication protocol	All EDAM modules	79
$AAP	Reads the communication protocol information	All EDAM modules	80
$AARS	Reset the module to power-on state	All EDAM modules	81
$AAS1	Reloads the module factory default	All EDAM modules	82
~AAX3IO	Set DIO LED ON/OFF Configuration	For 80xxD DIO modules	83
~AAX3	Read DIO LED ON/OFF	For 80xxD DIO modules	83

■ DIO Function Commands

Syntax	Description	Modules	Page
#**	Synchronized Sampling	For EDAM DIO modules	85
$AA4	Read synchronized data	For EDAM DIO modules	86
$AA6	Reads the Digital I/O Status (ref. "@AA")	For EDAM DIO modules	87
#AA00DD	Sets the digital output value of the lower eight channels (same as "#AA0ADD")	For 80xx D/O modules	89
#AA00DDDD	Sets the digital output value for channel(0~15), (same as ADAM-4000 "#AABB")	For 80xx D/O modules	90
#AA0ADD	Sets the digital output value of the lower eight channels. (same as "#AA00DD")	For 80xx D/O modules	91
#AA0BDD	Sets the digital output value of the upper eight channels	For 80xx D/O modules	92
#AA1CDD	Sets a single digital output for channel N (same as "#AAACDD" and “#AABCDD”)	For 80xx D/O modules	93
#AAACDD	Sets a single digital output channel of the lower eight channels.	For 80xx D/O modules	94
#AABCDD	Sets a single digital output channel of the upper eight channels.	For 80xx D/O modules	95
@AA	Reads the status of the digital input/output ports. (ref. "$AA6")	For EDAM DIO modules	96
@AA(data)	Sets the digital output channels	For 80xx D/O modules	97
#AAN	Reads the digital input counter of channel N.	For 80xx D/I modules	99
$AACN	Clears the digital input counter of channel N.	For 80xx D/I modules	100
~AAX4TT	Set debounce time for input counter	For 80xx D/I modules	101
~AAX4	Read input counter debounce time	For 80xx D/I modules	102
$AAC	Clear latched digital input	For 80xx D/I modules	103
$AALS	Read latched digital input	For 80xx D/I modules	104
~AADMN	Set DIO active status	For 80xx DI modules	105
~AAD	Read DIO active status	For 80xx DI modules	106

■ Watchdog Commands

Syntax	Description	Modules	Page
~**	Informs all modules that the host is OK	All EDAM modules	107
~AA0	Reads the host watchdog status of a module	All EDAM modules	108
~AA1	Resets the host watchdog timeout status of a module	All EDAM modules	109
~AA2	Read host watchdog timeout interval	All EDAM modules	110
~AA3EVV	Set Host Watchdog Timeout interval	All EDAM modules	111
~AA5V	Set Power-On & Safe Value of Channel N	All EDAM modules	113
~AA4V	Read Power-On & Safe Value of Channel N	All EDAM modules	115

4.5 ASCII Command Description

4.5.1 %AANNTTCCFF

Description:	Set module configuration	
Command:	%AANNTTCCFF[CHK](cr)	
Syntax:	%	Command leading code
	AA	Module address ID (00 to FF)
	NN	New address of the module (00 to FF)
	TT	Type code, should be hex 40 for DIO module
	CC	New baud rate code (ref. 3.13.1)
	FF	Data format (ref. 3.13.2)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	New Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note: When you want to change the checksum or baud rate, the **INIT*** pin must be grounded at first (see Appendix A),
or use Soft INIT* command (ref. ~AAI , ~AATnn).

Example(1): Change ID address from 01 to 03 (Assume current baud rate is 9600 and checksum disabled), response new module ID address 03 (change ID address only)

Command: %0103400600(cr)
Response: !03(cr)

Example(2): Change baud rate from 9600 to 19200(Assume current ID is 03, baud rate is 9600, and checksum disabled).

Because that the baud rate is changed from 9600 to 19200, the following procedures should be done before sending this command:

1. Power off the module
2. Short INIT* pin to Ground
3. Power on the module
4. Send command string
   Command: %0003400700(cr)
5. Response module ID address 03
   Response: !03(cr)
6. Power off module
7. Open **INIT*** pin and power on module again

Example(3): Enable checksum(Assume current ID is 03, baud rate is 9600 and checksum disabled).

Because that the checksum is changed from disable to enable, the following procedures should be done before sending this command:

1. Power off the module
2. Short INIT* pin to Ground (see Appendix A)
3. Power on the module
4. Send command string
   Command: %0003400640(cr)
5. Response module ID address 03
   Response: !03(cr)
6. Power off module
7. Open INIT* pin and power on module again (checksum enabled)

Example(4): Change baud rate from 9600 to 19200 and enable checksum (Assume current ID is 03, baud rate is 9600 and checksum disabled). Because that both the baud rate and checksum is changed , the following procedures should be done before sending this command:

1. Power off the module
2. Short **INIT*** pin to Ground (see Appendix A)
3. Power on the module
4. Send command string
   Command: %0003400740(cr)
5. Response module ID address 03
   Response: !03(cr)
6. response module ID address 03
7. Power off module
8. Open INIT* pin and power on module again (Baud rate changed to 19200 and checksum enabled)

It is recommended to use the setup utility to configure the module

Related topics: $AA2, ~AAI, ~AATnn

4.5.2 $AA2

Description:	Read module configuration	
Command:	$AA2[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	2	Command for reading configuration
	CHK	Check sum
	(cr)	Carriage return
Response:	!AATTCCFF[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	TT	Type code
	CC	Baud rate (ref. 3.13.1)
	FF	Data format of module (ref. 3.13.2)
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8050D(ID=01), Read configuration of module with ID address=01 and returns "400600"
(TT=40, baud rate=9600, no checksum)

Command: $012(cr)
Response : !01400600(cr)

Related command: %AANNTTCCFF

4.5.3 ~AAI

Description:	The Soft INIT* command is used to enable modification of the Baud Rate, checksum and communication protocol settings using software only. (The command is for firmware version D02.01 and later.)	
Command:	~AAI[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	I	Command to set the Soft INIT*
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note: The "**~AATnn"** command should be sent prior to sending this command.

Example : Sets the soft INIT* of module 01 and returns a valid response.

Command: ~01I(cr)
Response : !01(cr)

Related command: %AANNTTCCFF, ~AATnn, ~AAI, $AAPN

4.5.4 ~AATnn

Description:	Sets the soft INIT* timeout value. (The command is for firmware version D02.01 and later.)	
Command:	~AATnn[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	T	Command to set the soft INIT time out value
	nn	Two hexadecimal digits representing the timeout value in seconds. The maximum timeout value is 60 seconds. When changing the Baud Rate or checksum settings without altering the INIT* pin, the ~AAI and %AANNTTCCFF(or $AAPN) commands should be sent consecutively and the time interval between the two commands should be less than the soft INIT* timeout. If the soft INIT* timeout is 0, then the Baud Rate and checksum settings cannot be changed using software only. The power-on reset value of the soft INIT* timeout is 0.
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example(1) :

(1) Sets the soft INIT* of module 01 and returns a valid response.
Command: ~01I (cr) **Response:** !01(cr)

(2) Attempts to change the Baud Rate of module 01 to 19200 without first altering the INIT * pin. The module returns an invalid response because the soft INIT timeout value is 0.
Command: %0101000700 (cr) **Response:** ?01(cr)

(3) Sets the soft INIT* timeout value of module 01 to 16 seconds and returns a valid response.
Command: ~01T10 (cr) **Response:** !01(cr)

(4) Sets the soft INIT* of module 01 and returns a valid response.
Command: ~01I (cr) **Response:** !01(cr)

(5) Changes the Baud Rate of module 01 to 19200 without first altering INIT * pin. The module returns
Command: %0101000700 (cr) **Response:** !01(cr)

Example(2) :

(1) Sets the soft INIT* of module 01 and returns a valid response.
Command: ~01I (cr) **Response:** !01(cr)

(2) Attempts to change the protocol of module 01 to modbus-rtu without first altering the INIT * pin. The module returns an invalid response because the soft INIT timeout value is 0.
Command: $01P1 (cr) **Response:** ?01(cr)

(3) Sets the soft INIT* timeout value of module 01 to 16 seconds and returns a valid response.
Command: ~01T10 (cr) **Response:** !01(cr)

(4) Sets the soft INIT* of module 01 and returns a valid response.
Command: ~01I (cr) **Response:** !01(cr)

(5) Changes the protocol of module 01 to modbus-rtu without first altering INIT * pin. The module returns
Command: $01P1 (cr) **Response:** !01(cr)

Related command: %AANNTTCCFF, ~AAI, $AAPN

4.5.5 $AA5

Description:	Reads the Reset Status of a module	
Command:	$AA5[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	5	Command for read reset status
	CHK	Check sum
	(cr)	Carriage return
Response:	!AAS[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	S	= 0 - the module is not been reset
		= 1 - the module is been reset
	CHK	Check sum
	(cr)	Carriage return

Example : Read address 01 Read reset status and return module is been reseted

Command: $015(cr)
Response : !011(cr) - the module is been reseted

Command: $015(cr)
Response : !010(cr) - the module is not been reseted

Related command: $AARS

4.5.6 $AAF

Description:	Read the firmware version of a module	
Command:	$AAF[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	F	Command for Read Firmware Version
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	(data)	firmware version of module(max. 6 chars.)
	CHK	Check sum
	(cr)	Carriage return

Example : Read address 01 Read Firmware Version and return version D02.01

Command: $01F(cr)
Response: !01D02.01 (cr) - BIOS version D02.01

Related command:

4.5.7 $AAM

Description:	Read the module name	
Command:	$AAM[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	M	Command for Read Module Name
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	(data)	A string showing the name of the module (max. 6 chars.)
	CHK	Check sum
	(cr)	Carriage return

Example: Read name of module 01 and return the module name "8043"

Command: $01M(cr)
Response : !018043(cr)

Related command: ~AAO(data)

4.5.8 ~AAO(data)

Description:	Sets the module name	
Command:	~AAO(data)[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	O	Command to Sets the name of a module
	(data)	New name of the module (max. 6 characters).
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note: The new name is saved in the EEPROM

Example:

(1) Read name of module 01 and return the module name "8043"
Command: $01M(cr)
Response : !018043 (cr)

(2) Sets the name of the module 01 to be "EDAM" and returns a valid response.
Command: ~01OEDAM (cr)
Response : !01 (cr)

(3) Read address 01 Read Firmware Version, return the module name "EDAM"
Command: $01M(cr)
Response : !01EDAM(cr)

Related command: $AAM

4.5.9 $AAPN

Description:	Set the communication protocol (The command is for firmware version D02.01 and later.)	
Command:	$AAPN[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	P	Command to Set the communication protocol
	N	The protocols supported by the module = 0 - EDAM-ASCII format protocol (default) = 1 - Modbus-RTU protocol
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note:

1. Before the command is issued, the **INIT*** pin should be connected to GND or use Soft INIT* command (ref. ~AAI , ~AATnn).

2. The new protocol is saved in the EEPROM and will be effective after the next power on reset (Open **INIT*** pin).

Example : Sets the communication protocol of module 01 to Modbus-RTU and returns an valid response

Command: ~01P1 (cr)
Response : !01(cr)

Related command: ~AAP, ~AAI, ~AATnn

4.5.10 $AAP

Description:	Reads the communication protocol information (The command is for firmware version D02.01 and later.)	
Command:	$AAP[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	P	Command for Read protocol information
	CHK	Check sum
	(cr)	Carriage return
Response:	!AASC[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	S	The protocols supported by the module = 0 - Only ASCII protocol is supported = 1 - Both the EDAM ASCII and Modbus RTU protocols are supported
	C	The protocols supported by the module = 0 - EDAM-ASCII format protocol = 1 - Modbus-RTU protocol
	CHK	Check sum
	(cr)	Carriage return

Example : Reads the communication protocol of module 01 and returns a response of "10" meaning that it supports both the EDAM ASCII and Modbus RTU protocol and the protocol that will be used at the next power on reset is EDAM ASCII.

Command: $01P(cr)
Response : !0110(cr)

Related command: $AAPN

4.5.11 $AARS

Description:	Reset the module to power-on state (The command is for firmware version D02.01 and later.)	
Command:	$AARS[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	RS	Reset command
	CHK	Check sum
	(cr)	Carriage return
Response:	No response	

Note: Reset command will reset module to reboot.
(This command has no response from module)

Example : Reset module with ID address is 02

Command: $02RS (cr)
Response : No response

Related command: $AA5

4.5.12 $AAS1

Description:	Reloads the module factory default (The command is for firmware version D02.01 and later.)	
Command:	$AAS1[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	S1	Command to reload the factory default
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note: Before the command is issued, the **INIT*** pin should be connected to GND and after Response command is issued, the module will be rebooted.

Example: Reloads the module factory default setting and return valid.

Command: $05S1(cr)
Response : !05(cr)

Related command: %AANNTTCCFF, $AA2

4.5.13 ~AAX3IO

Description:	Set DIO module LED display panel on/off	
Command:	~AAX3IO[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	X3	Status LED control command.
	I	Input status LED control, = 1 - Turn-ON, if input active = 0 - Turn-OFF, if input active
	O	Output status LED control = 1 - Turn-ON LED, if output active = 0 - Turn-OFF LED, if output active
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example: Set module with ID=02 to turn-on the LED when relative input channels are active and output channels are active.

Command: ~02X311 (cr)
Response : !02 (cr)

Related command: ~AAX3

4.5.14 ~AAX3

Description:	Read status LED display panel control settings	
Command:	~AAX3[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	X3	Read LED setting command.
	CHK	Check sum
	(cr)	Carriage return
Response:	!AAIO[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	I	Input status LED control, = 1 - Turn-ON, if input active = 0 - Turn-OFF, if input active
	O	Output status LED control = 1 - Turn-ON LED, if output active = 0 - Turn-OFF LED, if output active
	CHK	Check sum
	(cr)	Carriage return

Example: Read LED control settings of module with ID=02.

Command: ~02X3 (cr)
Response : !0210 (cr)

Input LED will turn-on when input channels are active and output LED will turn-off when output channels are active.

Related command: ~AAX3IO

4.5.15 #**

Description:	Synchronize all modules to sample input values and store the values in the module's register at the same time and use "Read Synchronized Data" command to read the data and process it one by one.	
Command:	#**[CHK](cr)	
Syntax:	#	Command leading code
	**	Synchronized Sampling command
	CHK	Check sum
	(cr)	Carriage return
Response:	No response	

Example : Synchronized sampling command has no response

Command: #**<CR>
Response:

Related command: $AA4

4.5.16 $AA4

Description:	Read synchronized data	
Command:	$AA4[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	4	Command for reading synch. data
	CHK	Check sum
	(cr)	Carriage return
Response:	!SDDDD00[CHK](cr)	Valid command
	? AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	S	Data status, S=1 first read, S=0 been read
	DDDD	Data (4 characters) (ref. 3.14)
	00	The value is always 00
	CHK	Check sum
	(cr)	Carriage return

Example: Read Synchronized data from EDAM8053 (ID=05), and returns
S = 1 – first read, synchronized data=0978
(The first data DI8~DI15 = 09H, The second data DI0~DI7 = 78H)
Command: $054(cr)
Response: !1097800(cr)

Related command: #**

4.5.17 $AA6

Description:	Read the digital input channel value and read back the digital output channel value. (ref. "@AA")	
Command:	$AA6[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	6	Command for reading digital I/O status
	CHK	Check sum
	(cr)	Carriage return
Response:	!DDDD00[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	DDDD	A four-digit hexadecimal I/O value **(See 3.14)**
	00	The value is always 00
	CHK	Check sum
	(cr)	Carriage return

Example(1): For the EDAM-8050D(ID=02), Reads the digital input/output port status of module and returns 3A7Ch, which denotes that the first data (3A) DO1,DO3,DO4 and DO5 are ON(1) and the second data(7C) DI2,DI3,DI4, DI5 and DI6 are ON(1). **(See 3.14 and 3.15)**
Command: $026(cr)
Response: !3A7C00 (cr)

Example(2): For the EDAM-8042D(ID=05), Reads the digital output port status of module and returns 1A7Dh, which denotes that first data (1A) DO9,DO11 and DO12 are ON(1) and the second data(7D) DO0, DO2, DO3,DO4,DO5 and DO6 are ON(1).
Command: $056 (cr)
Response: !1A7D00(cr)

Example(3): For the EDAM-8060D(ID=02), Reads the digital input/output port status of module and returns 0F00h, which denotes that RL1,RL2, RL3 and RL4 are ON(1) and DI0,DI1, DI2 and DI3 are OFF(0).
Command: $026(cr)
Response: !0F0000 (cr)

Example(4): For the EDAM-8052D(ID=02), Reads the digital input port status of module and returns 7C00h, which denotes that the first data (7C) DI2,DI3,DI4, DI5 and DI6 are ON(1) and the second data(00). **(See 3.14 and 3.15)**
Command: $026(cr)
Response: !7C0000 (cr)

Example(5): For the EDAM-8041D(ID=05), Reads the digital input port status of module and returns 1A7Dh, which denotes that first data (1A) DI9, DI11 and DI12 are ON(1) and the second data(7D) DI0, DI2, DI3, DI4, DI5 and DI6 are ON(1).
Command: $056 (cr)
Response: !1A7D00(cr)

Related command: @AA, @AA(data) ,#AA0ADD, #AA0BDD

4.5.18 #AA00DD

Description:	Sets the digital output value of the lower eight channels (This command is the same with "#AA0ADD" command)	
Command:	#AA00DD [CHK](cr)	
Syntax:	#	Command leading code
	AA	Module address ID (00 to FF)
	00	Output command type
	DD	A two-digit hexadecimal value, where bit 0 corresponds to DO0, bit 1 corresponds to DO1, etc. When the bit is 1, it denotes that the digital output channel is ON, and 0 denotes that the digital output channel is OFF. (see sec. 3.14 DIO data format table)
	CHK	Check sum
	(cr)	Carriage return
Response:	>[CHK](cr)	Valid command
	?[CHK](cr)	Invalid command
	![CHK](cr)	Ignored command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	!	Delimiter for ignore command (The watchdog timeout status is set)
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8042D(ID=05), Sets DO1, DO3 and DO5 to ON , and DO0, DO2, DO4, DO6 and DO7 to OFF, and the module returns a valid response.
Command: #05002A<cr>
Response : >(cr)

Example: For the EDAM-8065D(ID=05), Sets RL2,RL4 and RL5 to ON , and RL1 and RL3 to OFF, and the module returns a valid response.
Command: #05001A<cr>
Response : >(cr)

Related command: #AA0ADD, #AA00DDDD, @AA(data)

4.5.19 #AA00DDDD

Description:	Sets the digital output value for channel(0~15), (This command is the for compatible with ADAM-4000 "#AABB")	
Command:	#AA00DDDD [CHK](cr)	
Syntax:	#	Command leading code
	AA	Module address ID (00 to FF)
	00	Output command type
	DDDD	A four-digit hexadecimal value, where bit 0 corresponds to DO0, bit 1 corresponds to DO1, etc. When the bit is 1, it denotes that the digital output channel is ON, and 0 denotes that the digital output channel is OFF. (see sec. 3.14 DIO data format table)
	CHK	Check sum
	(cr)	Carriage return
Response:	>[CHK](cr)	Valid command
	?[CHK](cr)	Invalid command
	![CHK](cr)	Ignored command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	!	Delimiter for ignore command (The watchdog timeout status is set)
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8043D(ID=05), Sets DO1, DO3,DO5,DO11 and DO12 to ON and the module returns a valid response.
Command: #0500182A<cr>
Response : >(cr)

Example: For the EDAM-8065D(ID=05), Sets RL2,RL4 and RL5 to ON and the module returns a valid response.
Command: #0500001A<cr>
Response : >(cr)

Related command: #AA0ADD, @AA(data) , #AA00DD

4.5.20 #AA0ADD

Description:	Sets the digital output value of the lower eight channels (This command is the same with "#AA00DD" command)	
Command:	#AA0ADD [CHK](cr)	
Syntax:	#	Command leading code
	AA	Module address ID (00 to FF)
	0A	Output command type
	DD	A two-digit hexadecimal value, where bit 0 corresponds to DO0, bit 1 corresponds to DO1, etc. When the bit is 1, it denotes that the digital output channel is ON, and 0 denotes that the digital output channel is OFF. (see sec. 3.14 DIO data format table)
	CHK	Check sum
	(cr)	Carriage return
Response:	>[CHK](cr)	Valid command
	?[CHK](cr)	Invalid command
	![CHK](cr)	Ignored command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	!	Delimiter for ignore command (The watchdog timeout status is set)
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8060D(ID=05), Sets RL2,RL3 to ON, and RL1, RL4 to OFF, and the module returns a valid response.
Command: #050A06<cr>
Response : >(cr)

Example: For the EDAM-8042D(ID=05), Sets DO1, DO3,DO4 and DO5 to ON , and DO0, DO2, DO4, DO6 and DO7 to OFF, and the module returns a valid response.
Command: #050A2A<cr>
Response : >(cr)

Related command: #AA00DD, @AA(data)

4.5.21 #AA0BDD

Description:	Sets the digital output value of the upper eight channels	
Command:	#AA0BDD [CHK](cr)	
Syntax:	#	Command leading code
	AA	Module address ID (00 to FF)
	0B	Output command type
	DD	A two-digit hexadecimal value, where bit 0 corresponds to DO8, bit 1 corresponds to DO9, etc. When the bit is 1, it denotes that the digital output channel is ON, and 0 denotes that the digital output channel is OFF. (see sec. 3.14 DIO data format table)
	CHK	Check sum
	(cr)	Carriage return
Response:	>[CHK](cr)	Valid command
	?[CHK](cr)	Invalid command
	![CHK](cr)	Ignored command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	!	Delimiter for ignore command (The watchdog timeout status is set)
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8043D(ID=05), Sets DO9, DO11 and DO12 to ON, and DO8, DO10,DO13, DO14 and DO15 to OFF, and the module returns a valid response.
Command: #050B1A<cr>
Response : >(cr)

Related command: #AA00DDDD, @AA(data)

4.5.22 #AA1CDD

Description:	Sets a single digital output for channel N (same as "AAACDD" and "#AABCDD")	
Command:	#AA1CDD [CHK](cr)	
Syntax:	#	Command leading code
	AA	Module address ID (00 to FF)
	1	Command to set a single digital output channel
	C	Specifies the digital output channel to be set (0 to F)
	DD	output ON/OFF state = 00 - set the digital output channel to OFF. = 01 - set the digital output channel to ON.
	CHK	Check sum
	(cr)	Carriage return
Response:	>[CHK](cr)	Valid command
	?[CHK](cr)	Invalid command
	![CHK](cr)	Ignored command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	!	Delimiter for ignore command (The watchdog timeout status is set)
	CHK	Check sum
	(cr)	Carriage return

Example(1): For the EDAM-8042D(ID=05), Set DO9 to ON, and the module returns a valid response.
Command: #051901<cr>
Response : >(cr)

Example(2): For the EDAM-8060D(ID=05), Set RL3 to OFF and the module returns a valid response.
Command: #051200<cr>
Response : >(cr)

Related command: #AAACDD, #AABCDD

4.5.23 #AAACDD

Description:	Sets a single digital output channel of the lower eight channels (same as "#AA1CDD")	
Command:	#AAACDD [CHK](cr)	
Syntax:	#	Command leading code
	AA	Module address ID (00 to FF)
	A	Output command type
	C	Specifies the digital output channel to be set (0 to 7)
	DD	output ON/OFF state = 00 - set the digital output channel to OFF. = 01 - set the digital output channel to ON.
	CHK	Check sum
	(cr)	Carriage return
Response:	>[CHK](cr)	Valid command
	?[CHK](cr)	Invalid command
	![CHK](cr)	Ignored command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	!	Delimiter for ignore command (The watchdog timeout status is set)
	CHK	Check sum
	(cr)	Carriage return

Example(1): For the EDAM-8043D(ID=05), Set DO2 to ON, and the module returns a valid response.
Command: #05A201<cr>
Response : >(cr)

Example(2): For the EDAM-8060D(ID=05), Set RL3 to OFF and the module returns a valid response.
Command: #05A200<cr>
Response : >(cr)

Related command: #AA1CDD , #AABCDD

4.5.24 #AABCDD

Description:	Sets a single digital output channel of the upper eight channels (same as "#AA1CDD")	
Command:	#AABCDD [CHK](cr)	
Syntax:	#	Command leading code
	AA	Module address ID (00 to FF)
	B	Output command type
	C	Specifies the digital output channel to be set (0 to 7) where 0 stands for channel 8, 1 stands for channel 9, etc.
	DD	output ON/OFF state = 00 - set the digital output channel to OFF. = 01 - set the digital output channel to ON.
	CHK	Check sum
	(cr)	Carriage return
Response:	>[CHK](cr)	Valid command
	?[CHK](cr)	Invalid command
	![CHK](cr)	Ignored command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	!	Delimiter for ignore command (The watchdog timeout status is set)
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8042D(ID=05), Set DO9 to ON, and the module returns a valid response.
Command: #05B101<cr>
Response : >(cr)

Related command: #AA1CDD , #AABCDD, #AAACDD

4.5.25 @AA

Description:	Reads the status of the digital input/output ports(ref. "$AA6")	
Command:	@AA [CHK](cr)	
Syntax:	@	Command leading code
	AA	Module address ID (00 to FF)
	CHK	Check sum
	(cr)	Carriage return
Response:	>DDDD[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	![CHK](cr)	Ignored command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	DDDD	A four-digit hexadecimal I/O value **(See 3.14)**
	CHK	Check sum
	(cr)	Carriage return

Example(1): For the EDAM-8050D(ID=02), Reads the digital input/output port status of module and returns 3A7Ch, which denotes that the first data (3A) DO1,DO3,DO4 and DO5 are ON(1) and the second data(7C) DI2,DI3,DI4,DI5 and DI6 are ON(1). **(See 3.14 and 3.15)**
Command: @02(cr)
Response: >3A7C(cr)

Example(2): For the EDAM-8042D(ID=05), Reads the digital output port status of module and returns 1A7Dh, which denotes that first data (1A) DO9,DO11 and DO12 are ON(1) and the second data(7D) DO0,DO2, DO3, DO4,DO5 and DO6 are ON(1).
Command: @056(cr)
Response: >1A7D(cr)

Example(3): For the EDAM-8060D(ID=02), Reads the digital input/output port status of module and returns 0F00h, which denotes that RL1, RL2, RL3 and RL4 are ON(1) and DI0,DI1, DI2 and DI3 are OFF(0).
Command: @02(cr)
Response: >0F00 (cr) **(See 3.14 and 3.15)**

Related command: $AA6, @AA(data), ~AADMN

4.5.26 @AA(data)

Description:	Sets the digital output channels	
Command:	@AA(data)[CHK](cr)	
Syntax:	@	Command leading code
	AA	Module address ID (00 to FF)
	(data)	Data to be written to the digital output channels. it is a one, two or four-digit hexadecimal value. where bit 0 corresponds to DO0, bit 1 corresponds to DO1, etc. When the bit is 1, it denotes that the digital output channel is ON, and 0 denotes that the digital output channel is OFF. **(See 3.14 and 3.15)** (data) = 0~F (one character) for 8060 (data) = 0~7 (one character) for 8063/A/B (data) = 00~FF (two characters) for 8044,8050 (data) = 00~1F (two characters) for 8065/A/B (data) = 00~FF (two characters) for 8066,8067 (data) = 0000~1FFF (four characters) for 8042 (data) = 0000~FFFF (four characters) for 8043,45
	CHK	Check sum
	(cr)	Carriage return
Response:	>[CHK](cr)	Valid command
	?[CHK](cr)	Invalid command
	![CHK](cr)	Ignored command
	>	Delimiter for valid command
	?	Delimiter for invalid command
	!	Delimiter for ignore command (The watchdog timeout status is set)
	CHK	Check sum
	(cr)	Carriage return

Example(1): For the EDAM-8060D(ID=05), Sets RL2,RL4 to ON, and RL1,RL3 to OFF, and the module returns a valid response.
Command: @05A<cr>
Response : >(cr)

Example(2): For the EDAM-8050D(ID=03), Sets DO2,DO3,DO5,DO6 to ON, and DO0,DO1,DO4,DO7 to OFF, and the module returns a valid response.
Command: @036C<cr>
Response : >(cr)

Example(3): For the EDAM-8043D(ID=05), Sets DO2,DO3,DO4, DO8, DO9, D13,D15 to ON, and DO0,DO1,DO5,DO6,DO7,DO10,DO11, DO12,DO14 to OFF, and the module returns a valid response.
Command: @05A31C<cr>
Response : >(cr)

Related command: #AA0ADD, #AA0BDD, @AA(data), #AA00DD

4.5.27 #AAN

Description:	Reads the digital input counter of channel N	
Command:	#AAN[CHK](cr)	
Syntax:	#	Command leading code
	AA	Module address ID (00 to FF)
	N	Digital input channel number (0~F)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AADDDDD[CHK](cr)	Valid command
	? AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	DDDDD	Five decimal digits representing the digital input counter data of the specified channel(00000 to 65535)
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8050D(ID=02), Read counter value of channel 5 and the returned counter value is 00245.
Command: #025(cr)
Response: !0200245(cr)

Ref. command: $AACN, ~AAX4TT, ~AAX4

4.5.28 $AACN

Description:	Clears the digital input counter of channel N	
Command:	$AACN[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	C	Command for clears the digital input counter
	N	Digital input channel number (0~F)
	CHK	Check sum
	(cr)	Carriage return
Response: (see Note)	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8050D(ID=06),Clear the digital input counter of channel 5.
Command: $06C5(cr)
Response: !06(cr)

Ref. command: #AAN, ~AAX4TT, ~AAX4

4.5.29 ~AAX4TT

Description:	Set debounce time for input counter	
Command:	~AAX4TT [CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	X4	Set debounce time command.
	TT	Debounce time value(01~FF) in hexadecimal format, Each count is 2ms. (01=2ms and FF=510ms, default 10ms debounce time)
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8050D(ID=06), Set debounce time to 50ms
Command: ~06X419(cr)
Response: !06(cr)

Ref. command: ~AA2X04, $AACN, #AAN

4.5.30 ~AAX4

Description:	Read input counter debounce time	
Command:	~AAX4 [CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	X4	Read debounce time command.
	CHK	Check sum
	(cr)	Carriage return
Response:	!AATT[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	TT	Debounce time value(01~FF) in hexadecimal format, Each count is 2ms. (01=2ms and FF=510ms)
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8050D(ID=06), Read debounce time and return 19(50ms).
Command: ~06X4 (cr)
Response: !0619(cr)

Ref. command: ~AAX4TT, $AACN, #AAN

4.5.31 $AAC

Description:	Clears the status of the latched digital input channels	
Command:	$AAC[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	C	Command for clearing latched digital input
	CHK	Check sum
	(cr)	Carriage return
Response: (see Note)	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example: Clear latched input of module address ID=06
Command: $06C<CR>
Response: !06<CR>

Ref. command: $AALS

4.5.32 $AALS

Description:	Reads the status of the latched digital input channels	
Command:	$AALS[CHK](cr)	
Syntax:	$	Command leading code
	AA	Module address ID (00 to FF)
	L	Command to read the latched status
	S	= 0 - Read the low latched status = 1 - Read the high latched status
	CHK	Check sum
	(cr)	Carriage return
Response: (see Note)	!DDDD00[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	DDDD	Status of the latched digital input channels, a four digit hexadecimal value. **(See 3.14)**
	00	The value is always 00
	CHK	Check sum
	(cr)	Carriage return

Example: For the EDAM-8053D(ID=06), read the status of the low latched digital input channels of module and returns FFFF.
Command: $06L0(cr)
Response: !06FFFF00(cr)

Sends the command to clear the status of the latched digital input channels of module 06 and returns a valid response.
Command: $06C(cr)
Response: !06(cr)

Read the status of the low latched digital input channels of module 06 and returns 0000.
Command: $06L0(cr)
Response: !06000000(cr)

Ref. command: $AAC

4.5.33 ~AADMN

Description:	Set DIO active status	
Command:	~AADMN[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	D	Command to Set DIO active status
	M	digital input channel active values = 0 - input value 0(OFF) for input active,. = 1 - input value 1(ON) for input active.
	N	digital output channel active values = 0 - output value 0(OFF) for output active = 1 - output value 1(ON) for output active
	CHK	Check sum
	(cr)	Carriage return
Response:	!AA[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example:

Read EDAM-8050D active status , Response: DI inactive read value is 0
Command: ~06D(cr)
Response: !0601(cr)
For the EDAM-8050D(ID=06),Reads the status of the DIO
Command: @06(cr)
Response: >01F2 (cr)
For the EDAM-8050D(ID=06), Set input active value to 1
Command: ~06D11(cr)
Response: !06 (cr)
For the EDAM-8050D(ID=06),Reads the status of the DIO
Command: @06(cr)
Response: >010D (cr)

Ref. command: ~AAD, @AA

4.5.34 ~AAD

Description:	Read DIO active status	
Command:	~AAD[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	D	Command to read DIO active status
	CHK	Check sum
	(cr)	Carriage return
Response:	!AAMN[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	M	digital input channel active values = 0 - input value 0(OFF) for input active,. = 1 - input value 1(ON) for input active.
	N	digital output channel active values = 0 - output value 0(OFF) for output active = 1 - output value 1(ON) for output active
	CHK	Check sum
	(cr)	Carriage return

Example:

For the EDAM-8050D(ID=06), Set input active value to 1(ON) and output active value to 0()FF).

Command: ~06D10(cr)
Response: !06 (cr)

Read input inactive status
Command: ~06D(cr)
Response: !0610(cr)

Ref. command: ~AADMN, @AA

4.5.35 ~**

Description:	Host send this command to all modules for send the information "Host OK"	
Command:	~**[CHK](cr)	
Syntax:	~	Command leading code
	**	For all modules
	CHK	Check sum
	(cr)	Carriage return
Response:	No response	

*: When host watchdog timer is enable, host computer must send this command to all module before timeout otherwise "Host watchdog timer enabled" module's output value will go to safety state output value.

Ref. command: ~AA0, ~AA1, ~AA2, ~AA3EVV, ~AA4V, ~AA5V

4.5.36 ~AA0

Description:	Read watchdog timeout status	
Command:	~AA0[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	0	Command for reading timeout status
	CHK	Check sum
	(cr)	Carriage return
Response:	! AASS[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	SS	Two hexadecimal digits that represent the host watchdog status. bit(7) - Host watchdog enable/disable , = 0 - Disable. = 1 - Enable. bit(2) - Host watchdog timeout status, = 0 - indicates that no host watchdog timeout has occurred. = 1 - indicates that a host watchdog timeout has occurred. bit(6,5,4,3,1,0) - reserved(=0) The host watchdog status is stored in EEPROM and can only be reset by using the ~AA1 command.
	CHK	Check sum
	(cr)	Carriage return

Example(1): Reads the host watchdog status of module 02 and returns 00, meaning that the host watchdog is disabled and no host watchdog timeout has occurred.
Command: ~020(cr)
Response: !0200(cr)

Example: (2) Reads the host watchdog status of module 02 and returns 04, meaning that a host watchdog timeout has occurred.
Command: ~020(cr)
Response: !0204(cr)

Ref. command: ~AA1, ~AA2, ~AA3EVV, ~AA4V, ~AA5V

4.5.37 ~AA1

Description:	Reset watchdog timeout status	
Command:	~AA1[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	1	Command for resetting watchdog timeout status
	CHK	Check sum
	(cr)	Carriage return
Response:	! AA [CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example: Reads the host watchdog status of module 03 and shows that a host watchdog timeout has occurred.
Command: ~030 (cr)
Response: !0304 (cr)

Resets the host watchdog timeout status of module 03 and returns a valid response.
Command: ~031 (cr)
Response: !03 (cr)

Reads the host watchdog status of module 03 and shows that no host watchdog timeout has occurred.
Command: ~030 (cr)
Response: !0300 (cr)

Ref. command: ~AA0, ~AA2, ~AA3EVV, ~AA4V, ~AA5V

4.5.38 ~AA2

Description:	Read host watchdog timeout value	
Command:	~AA2[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	2	Command for reading watchdog timeout value
	CHK	Check sum
	(cr)	Carriage return
Response:	! AAEVV[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	E	Host watchdog enabled status E = 1 – Enable E = 0 – Disable
	VV	Timeout value in hex format from 01 to FF (01 denotes 0.1 seconds and FF denotes 25.5 seconds)
	CHK	Check sum
	(cr)	Carriage return

Example: Reads the host watchdog timeout value of module 03 and returns FF, which denotes that the host watchdog is enabled and the host watchdog timeout value is 25.5 seconds.
Command: ~032 (cr)
Response: !031FF(cr)

Ref. command: ~AA0, ~AA1, ~AA3EVV, ~AA4V, ~AA5V

4.5.39 ~AA3EVV

Description:	Enables/disables the host watchdog and sets the host watchdog timeout value of a module.	
Command:	~AA3EVV[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	3	Command for setting watchdog timeout value
	E	1= enable, 0= disable Host watchdog
	VV	Timeout value (01~FF, each for 0.1 second)
	CHK	Check sum
	(cr)	Carriage return
Response:	! AA [CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Note:

If host watchdog timer is enabled, the host should send *Host OK* (see "~**") command periodically within Timeout value to refresh the timer, otherwise the module will be forced to safety state (see "~AA5V")

Example:

Set module (ID=04) to have watchdog timeout value 20.0 seconds and enable host watchdog.

Command: ~0431C8(cr)
Response: !04(cr)

Read watchdog timeout value form module (ID=04).
The module returns 1C8, which denotes that the host watchdog is enabled and the host watchdog timeout value is 20.0 seconds.

Command: ~042(cr)
Response: !041C8(cr)

Host send this command to all modules for send the information "Host OK"

Command: ~**(cr)

Stop sending any command string to modules for at least 20.0 seconds. The LED on the module will go to flash. The flash LED indicates the host watchdog is timeout and timeout status is set.

Read watchdog timeout status, The module returns 04, which denotes that a host watchdog timeout has occurred.
Command: ~040(cr)
Response: !0404(cr)

Reset watchdog timeout status. Watchdog timeout is cleared and LED stop flashing, and host watchdog is disabled
Command: ~041(cr)
Response: !04 (cr)

Reads the host watchdog status of module 04 and returns 00, meaning that the host watchdog is disabled and no host watchdog timeout has occurred.
Command: ~040(cr)
Response: !0400(cr) Timeout status is cleared

Ref. command: ~AA0, ~AA1, ~AA2, ~AA4V, ~AA5V

4.5.40 ~AA5V

Description:	Sets the current DO value as the power-on DO value or the safe DO value.	
Command:	~AA5V[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	5	Command for Sets power on and safe value
	V	V= P – Set power on value V= S – Set safe value
	CHK	Check sum
	(cr)	Carriage return
Response:	! AA [CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	CHK	Check sum
	(cr)	Carriage return

Example(1):

For the EDAM-8050D(ID=04), Set module to have output value AA.
Command: @04AA(cr)
Response: > (cr)

For the EDAM-8050D(ID=04), Set current output value AA as safe value.
Command: ~045S(cr)
Response: !04(cr)

For the EDAM-8050D(ID=04), Set module to have output value 55.
Command: @0455(cr)
Response: > (cr)

For the EDAM-8050D(ID=04), Set current output value 55 as power-on value.
Command: ~045P(cr)
Response: !04(cr)

For the EDAM-8050D(ID=04), Read Power on value and return power-on value 55.
Command: ~044P (cr)
Response: !045500 (cr)

For the EDAM-8050D(ID=04), Read Power on value and return safe value AA.
Command: ~044S (cr)
Response: !04AA00 (cr)

Example(2):

For the EDAM-8043D(ID=04), Set module to have output value 55AA.
Command: @0455AA(cr)
Response: > (cr)

For the EDAM-8043(ID=04), Set current output value 55AA as safe value.
Command: ~045S(cr)
Response: !04(cr)

For the EDAM-8043D(ID=04), Set module to have output value 5A5A.
Command: @045A5A(cr)
Response: > (cr)

For the EDAM-8043D(ID=04), Set current output value 5A5A as power-on value.
Command: ~045P(cr)
Response: !04(cr)

For the EDAM-8043D(ID=04), Read Power on value and return power-on value 5A5A.
Command: ~044P(cr)
Response: !045A5A(cr)

For the EDAM-8050D(ID=04), Read Power on value and return safe value 55AA.
Command: ~044S (cr)
Response: !0455AA(cr)

Ref. command: ~AA0, ~AA1, ~AA2, ~AA3EVV, ~AA4V

4.5.41 ~AA4V

Description:	Reads the power-on DO value or the safe DO value of a module	
Command:	~AA4V[CHK](cr)	
Syntax:	~	Command leading code
	AA	Module address ID (00 to FF)
	4	Command for reading power on and safe value
	V	V= P – Read power on value V= S – Read safe value
	CHK	Check sum
	(cr)	Carriage return
Response:	! AA(data)[CHK](cr)	Valid command
	?AA[CHK](cr)	Invalid command
	!	Delimiter for valid command
	?	Delimiter for invalid command
	AA	Module address ID
	(data)	For the EDAM-8042D/8043D modules, they are four hexadecimal digits. For other modules, they are two hexadecimal digits followed by 00. (data) = xxxx - for EDAM-8042D, 8043D (data) = xx00 - for other modules
	CHK	Check sum
	(cr)	Carriage return

Example:

For the EDAM-8043D(ID=04), Read Power on value and return power-on value 5A5A.
Command: ~044P(cr)
Response: !045A5A(cr)

For the EDAM-8050D(ID=04), Read Power on value and return safe value AA.
Command: ~044S (cr)
Response: !04AA00 (cr)

Ref. command: ~AA0, ~AA1, ~AA2, ~AA3EVV, ~AA5V

Chapter 5 Modbus RTU Protocol Command Sets

5.1 Introduction

MODBUS Protocol is a messaging structure developed by Modicon in 1979, used to establish master-slave/client-server communication between intelligent devices. Detailed information can be found at http://www.modbus.org to find more valuable information.
E-8000 series modules are supported the Modbus RTU protocol **by firmware version D02.01 and later**.

The communication BaudRates rang from 1200bps to 115200bps. The parity, data bits and stop bits are fixed as no parity, 8 data bits and 1 stop bit. The following Modbus functions are supported.

5.2 MODBUS Data model

MODBUS bases its data model on a series of tables that have distinguishing characteristics.

The four primary tables are:

Primary	Object type	Type of	Comments
Discrete Input	Single bit	Read-Only	This type of data can be provided by an I/O system.
Coils	Single bit	Read-Write	This type of data can be alterable by an application
Input Reg.	16 bit word	Read-Only	This type of data can be provided by an I/O system
Holding Reg.	16 bit word	Read-Write	This type of data can be alterable by an application program.

5.3 MODBUS function code definition:

Function Code	Description
01 (0x01)	Read coils
02 (0x02)	Read Discrete Inputs
03 (0x03)	Read multiple Holding registers
04 (0x04)	Read multiple input registers
05 (0x05)	Write single coil
06 (0x06)	Write single register
15 (0x0F)	write Multiple coils
16 (0x10)	Write Multiple register
70 (0x46)	Read / write module settings

Error Response:

If the function specified in the message is not supported, then the module Response as follows:

Offset	Function	Length	Description
00	Address	1 Byte	1 to 247
01	Function code	1 Byte	Function code | 0x80
02	Exception code	1 Byte	01

If a CRC mismatch occurs, the module will not respond.
(ref. adde. "02208")

5.4 MODBUS Standard Register Designation

0xxxx - Coils access, (for 0x01, 0x05, 0x0F function code)
1xxxx - Read discrete inputs, (for 0x02 function code)
3xxxx - Read input register, (for 0x04 function code)
4xxxx - Holding register access, (for 0x03, 0x06, 0x10 function code)

xxxx - Element address of a data block, In the MODBUS data model each element within a data block is numbered from 1 to n.

Example:
00005 - Means Coils access and Starting address = 0004 (0005-1)
10002 - Means Read discrete inputs and Starting address = 0001 (0002-1)
30257 - Means Read input register and Starting address = 0256 (0257-1)
40001 - Means Access holding register and Starting address = 0000 (0001-1)

5.5 Modbus Address Mapping Table

There are three categories of EDAM-modules commands. The first is the General Commands, second is the DIO Function Commands and The third is the Watchdog Commands Sets. All the commands used in the EDAM DIO Input/Output module are list in the following table.

5.5.1 General Commands

Address Mapping			
Address	Item	Attr.	Sec.
00257	Protocol, EDAM ASCII & Modbus select. = 1 (0xFF00) - Modbus RTU = 0 (0x0000) - EDAM ASCII	R/W	5.5.5.1
00272	Load factory calibration parameters = 0xFF00 - Enable = 0x0000 - Disable	W	5.5.5.2
02208	CRC checking enable / disable = 1 (0xFF00) - Enable = 0 (0x0000) - Disable (default)	R/W	5.5.5.3
02210	Reset the module to initial power-on status = 0xFF00 - Enable = 0x0000 - Disable	W	**5.5.5.4**
00273, 10273	Read module reset status = 1 - first read after powered on = 0 - not the first read after powered on	R	**5.5.5.5**
40481	Firmware version (low word)	R	5.5.5.6
40482	Firmware version (high word)	R	5.5.5.6
40483	Module name (low word)	R	5.5.5.7
40484	Module name (high word)	R	5.5.5.7
40485	Module address, valid range: 1 ~ 247	R/W	5.5.5.8
40486	Baudrate setting, valid range: 3 ~ 10 for Baudrate (1200,2400,4800,9600,19200,38400,57600,115200)	R/W	3.13.1
40488	Modbus response delay time in ms (0~30ms)	R/W	5.5.5.10

5.5.2 Watchdog Commands

Address Mapping			
Address	Item	Attr.	Sec.
412345, 312345	Informs all modules that the host is OK (for 8042,8043,8044,8050, 8060,8063,8065,8066, 8067)	R	5.5.6
40489	Host watchdog timeout value, 0 ~ 255, in 0.1second	R/W	5.5.6
40492	watchdog timeout counter, write 0 to reset counter.	R/W	5.5.6
00261	Host watchdog enable/disable, = 1 (0xFF00) - Enable = 0 (0x0000) - Disable (for 8042,8043,8044,8050,8060,8063,8065,8066, 8067)	R/W	5.5.6
00270	Host watch dog timeout status, write 1(0xFF00) to clear host watchdog timeout status.	R/W	5.5.6

5.5.3 DIO Function Commands

Address Mapping			
Address	Item	Attr.	Sec.
00001~00032	Digital output channel for DO0~DO31 (for 8042,8043,8044,8050,8060,8063,8065,8066, 8067)	R/W	5.5.7.1
00033~00064	Digital input channel for DI0~DI31 (for 8041,8052,8053,8044, 8050, 8060, 8063,8065)	R	5.5.7.2
10001~10032	Digital input channel for DI0~DI31 (for 8041,8052,8053,8044, 8050, 8060, 8063, 8065)	R	5.5.7.2
00065~00096	DI Latch high value for DI0~DI31 (for 8041,8052,8053,8044, 8050, 8060, 8063, 8065)	R	5.5.7.3
00097~00128	DI Latch low value for DI0~DI31 (for 8041,8052,8053,8044, 8050, 8060, 8063, 8065)	R	5.5.7.3
00264	Clear the latch value , = 1 (0xFF00) - Cleare (for 8041,8052,8053,8044, 8050, 8060, 8063, 8065)	W	5.5.7.3
30001~30032	Digital input counter for DI0~DI31 (for 8041,8052,8053,8044, 8050, 8060, 8063, 8065)	R	5.5.7.4
00513~00544	Clear the DI counter value for DI0~DI31, = 1 (0xFF00) - Cleare (for 8041,8052,8053,8044, 8050, 8060, 8063, 8065)	W	5.5.7.4
02251	Digital Input Count Edge, 1st byte contains the DI0~DI7 and the second byte contains DI8~DI15,..... = 1 (0xFF00) - Rising edge = 0 (0x0000) - falling edge (for 8041,8052,8053,8044, 8050, 8060, 8063, 8065)	R/W	5.5.7.4
00129~00160	Safe value for DO0~DO31 (for 8042,8043,8044,8050, 8060,8063,8065,8066, 8067)	R/W	5.5.7.5
00161~00192	Power-on value for DO0~DO31 (for 8042,8043,8044,8050, 8060,8063,8065,8066, 8067)	R/W	5.5.7.6
42201	Digital input/output LED Configuration (for 80xxD) bit(1) - for digital output LED control: = 0 - Turn-ON LED when output active = 1 - Turn-ON LED when output inactive bit(0) - for digital input LED control: = 0 -Turn-ON LED when input low (active) =1 -Turn-ONLED when input high(inactive)	R/W	5.5.7.7

Address	Item	Attr.	Sec.
42209	DIO active status, bit(0) - DI active status (IAS): = 0 - input value 1 for non-signal or high(open).; input value 0 for low(GND). = 1 - input value 0 for non-signal or high(open).; input value 1 for low(GND). bit(1) - DO active status(OAS): = 0 - output value 1 for output relay active, output value 0 for output relay inactive. = 1- output value 0 for output relay active, output value 1 for output relay inactive.	R/W	5.5.7.8

5.5.4 70 (0x46) Read/Write Module Command Sets

This function code is used to read the settings of the module or change the settings of the module.

The following sub-function codes are supported:

Sub-Function code	Description	Sec.
00 (0x00)	Read module name	*5.5.8.1*
04 (0x04)	Set the module address	*5.5.8.2*
05 (0x05)	Read the communication settings	5.5.8.3
06 (0x06)	Set the BaudRate, communication protocol and CRC check	5.5.8.4
32 (0x20)	Read the firmware version	5.5.8.5
33 (0x21)	Set Digital Input Count Edge	5.5.8.6
34 (0x22)	Read Digital Input Count Edge	5.5.8.7
39 (0x27)	Set Power-On output value	5.5.8.8
40 (0x28)	Read Power-On output value	5.5.8.9
41 (0x29)	Set DIO active status	5.5.8.10
42 (0x2A)	Read DIO active status	5.5.8.11
53 (0x35)	Read the response delay time	5.5.8.13
54 (0x36)	Set the response delay time	5.5.8.12
128 (0x80)	Set Digital input/output LED Configuration	5.5.8.14
129 (0x81)	Read Digital input/output LED Configuration	5.5.8.15

Error Response:

If the function specified in the message is not supported, then the module Response as follows:

Offset	Function	Length	Description
00	Address	1 Byte	1 to 247
01	Function code	1 Byte	0xC6
02	Exception code	1 Byte	= 02 - invalid sub-function code

If a CRC mismatch occurs, the module will not respond.

5.5.5 Example of modbus RTU General commands

5.5.5.1 (00257) Protocol, EDAM ASCII & Modbus select

- (00257) Read protocol and return modbus RTU is select (01).
  Request : 01 01 01 00 00 01 [FC 36]
  Response: 01 01 01 01 [90 48]

- (00257) Set to EDAM ASCII protocol and return successful
  Request : 01 05 01 00 00 00 [CC 36]
  Response: 01 05 01 00 00 00 [CC 36]

- (00257) Read protocol and return (0x00) EDAM ASCII is select
  Request : 01 01 01 00 00 01 [FC 36]
  Response: 01 01 01 00 [51 88]

5.5.5.2 (00272)Load factory calibration parameters

- (00272) Load factory calibration parameters and wait 1sec for response.
  Request : 01 05 01 0F FF 00 [BD C5]
  Response: 01 05 01 0F FF 00 [BD C5]

5.5.5.3 (02208)CRC checking status

(02208) Read CRC checking status and return CRC disable (00)
Request : 01 01 08 9F 00 01 [CF 84]
Response: 01 01 01 00 [51 88]

- (02208) Set CRC checking to enable and return successful
  Request : 01 05 08 9F FF 00 [BE 74]
  Response: 01 05 08 9F FF 00 [BE 74]

- (02208) Set CRC checking to disable and return successful
  Request : 01 0F 08 9F 00 01 01 00 [BB C3]
  Response: 01 0F 08 9F 00 00 [A6 45]

5.5.5.4 (02210)Reset the module to initial power-on status

- (02210) Reset the module to initial power-on status and return successful
  Request : 01 05 08 A1 FF 00 [DF B8]
  Response: 01 05 08 A1 FF 00 [DF B8]

5.5.5.5 (00273)Read module reset status

- (00273) Read module reset status and return first read after powered on (01).
  Request : 01 01 01 10 00 01 [FD F3]
  Response: 01 01 01 01 [90 48]

- (00273) Read module reset status and return not first read (00).
  Request : 01 02 01 10 00 01 [B9 F3]
  Response: 01 02 01 00 [A1 88]

5.5.5.6 (40481)Read Firmware version

- (40481) Read firmware version and return version D02.01(00 0D 02 01)
  Request : 01 03 01 E0 00 02 [C4 01]
  Response: 01 03 04 00 0D 02 01 [AB 50]

5.5.5.7 (40483)Module name

- (40483) Read Module name and return module name 8050(00 80 50 00)
  Request : 01 03 01 E2 00 02 [65 C1]
  Response: 01 03 04 00 80 50 00 [C7 DB]

5.5.5.8 (40485)Module address

- (40485) Read Module address and return module address 01 (00 01)
  Request : 01 03 01 E4 00 01[C5 C1]
  Response: 01 03 02 00 01[79 84]

- (40485) Set new module address to 05 and return successful
  Request : 01 06 01 E4 00 05 [08 02]
  Response: 01 06 01 E4 00 05 [08 02]

- (40485) Read module address and return module address 01 (00 01)
  Request : 01 03 01 E4 00 01[C5 C1]
  Response: 01 03 02 00 05[78 47]

- (40485) Set new module address to 01 and return successful
  Request : 01 10 01 E4 00 01 02 00 01 [60 B4]
  Response: 01 10 01 E4 00 01 [40 02]

5.5.5.9 (40486) Baudrate setting

- (40486) Read baudrate and return baudrate 9600 (00 06)
  Request : 01 03 01 E5 00 01 [94 01]
  Response: 01 03 02 00 06 [38 46]

- (40486) Set baudrate to 115200(0A) and return successful
  (the **INIT*** pin must be grounded at first)
  Request : 01 06 01 E5 00 0A [19 C6]
  Response: 01 06 01 E5 00 0A [19 C6]

- (40486) Read baudrate and return baudrate 115200 (00 0A)
  Request : 01 03 01 E5 00 01 [94 01]
  Response: 01 03 02 00 0A [38 43]

- (40486) Set baudrate to 115200(0A) and return successful
  (the **INIT*** pin must be grounded at first)
  Request : 01 10 01 E5 00 01 02 00 06 [20 A7]
  Response: 01 10 01 E5 00 01 [11 C2]

5.5.5.10 (40488) Modbus response delay time in ms

- (40488) Set modbus response delay time to 37ms and return successful
  Request : 01 06 01 E7 00 25 [F9 DA]
  Response: 01 06 01 E7 00 25 [F9 DA]

- (40488) Read modbus response delay time and return 37ms(00 25)
  Request : 01 03 01 E7 00 01 [35 C1]
  Response: 01 03 02 00 25 [79 9F]

- (40488) Set modbus response delay time to 0ms and return successful
  Request : 01 10 01 E7 00 01 02 00 00 [A1 47]
  Response: 01 10 01 E7 00 01 [B0 02]

5.5.6 Example of modbus RTU Watchdog commands

Host watchdog timeout operation

- (00129) Set output channel(0,2,4,5,9) to ON(0000 0010 0011 0101) for write safe value and return successful.
  Request : 01 0F 00 80 00 0A 02 35 02 [6C 69]
  Response: 01 0F 00 80 00 0A [D4 24]

- (40489) Write host watchdog timeout value(20 sec)
  return valid.
  Request : 01 06 01 E8 00 C8 [09 94]
  Response: 01 06 01 E8 00 C8 [09 94]

- (40489) Read host watchdog timeout value
  return (00 C8) watchdog timeout value(20 sec).
  Request : 01 03 01 E8 00 01 [05 C2]
  Response: 01 03 02 00 C8 [B9 D2]

- (00001) Set DO output channel(0~12) to 0 and return successful.
  Request : 01 0F 00 00 00 0D 02 00 00 [E4 4C]
  Response: 01 0F 00 00 00 0D [94 0E]

- (00261) Set host watchdog timeout enable,
  return valid.
  Request : 01 05 01 04 FF 00 [CC 07]
  Response: 01 05 01 04 FF 00 [CC 07]
  ;
  wait 15 sec……
  ;
- (00270) Clear host watchdog timeout counter,
  return watchdog timeout is set.
  Request : 01 05 01 0D FF 00 [1C 05]
  Response: 01 05 01 0D FF 00 [1C 05]
  ;
  wait 15 sec……
  ;
- (412345) Informs all modules that the host is OK and no response
  Request : 01 04 30 38 00 00 [7E C7]
  Response: no response
  ;
  wait 15 sec……
  ;
- (312345) Informs all modules that the host is OK and no response
  Request : 01 03 30 38 00 00 [CB 07]
  Response: no response
  ;
  wait 25 sec……
  watchdog timeout and into safe output mode

- (00270) Read host watchdog timeout status,
  return host watchdog timeout flag is set.
  Request : 01 01 01 0D 00 01 [6D F5]
  Response: 01 01 01 01 [90 48]

- (00001) Read output channel(0~12) and return safe value(2F 6D)
  Request : 01 01 00 00 00 0D [FD CF]
  Response: 01 01 02 35 02 [2F 6D]

- (00270) Clear host watch dog timeout status,
  return host watchdog timeout flag is set.
  Request : 01 05 01 0D FF 00 [1C 05]
  Response: 01 05 01 0D FF 00 [1C 05]

5.5.7 Example of modbus RTU DIO Function Commands

5.5.7.1 (00001) Digital output channel for DO0~DO31

- (00001) Set output channel(0,2,4,5,9) to ON(0000 0010 0011 0101)
  and return successful.
  Request : 01 0F 00 00 00 0D 02 35 02 [E4 4C]
  Response: 01 0F 00 00 00 0D [94 0E]

- (00001) Read output channel(0~12) and return safe value(35 03)
  Request : 01 01 00 00 00 0D [FD CF]
  Response: 01 01 02 35 02 [2F 6D]

- (00001) Set DO11 output channel ON and return successful.
  Request : 01 05 00 0B FF 00 [FD F8]
  Response: 01 05 00 0B 00 0D [FD F8]

- (00001) Read output channel(11) and return (01) ON
  Request : 01 01 00 0B 00 01 [8C 08]
  Response: 01 01 01 01 [90 48]

5.5.7.2 (00033 & 10001) Digital input channel for DO0~DO31

- (00033) Read input channel(0~14) and return all ON(FF 3F).
  Request : 01 01 00 20 00 0E [BC 04]
  Response: 01 01 02 FF 3F [B8 1C]

- (10001) Read output channel(0~12) and return all ON(FF 3F).
  Request : 01 02 00 00 00 0E [F9 CE]
  Response: 01 02 02 FF 3F [B8 58]

5.5.7.3 (00065, 00097,00264) DI Latch for DI0~DI31

- (00065) Read DI(0~14) Latch high value and return (FF 3F).
  Request : 01 01 00 40 00 0E [BC 1A]
  Response: 01 01 02 FF 3F [B8 1C]

- (00097) Read DI(0~14) Latch low value and return (00 00).
  Request : 01 01 00 60 00 0E [BD D0]
  Response: 01 01 02 00 00 [B9 FC]

- (00264) Clear channel(0) latch value and return successful.
  Request : 01 05 01 07 FF 00 [3C 07]
  Response: 01 05 01 07 FF 00 [3C 07]

5.5.7.4 (30001) Digital input counter for DI0~DI31

- (30001)Read Digital input counter for DI1~DI2 and return(00 00 00 00).
  Request : 01 04 00 01 00 02 [20 0B]
  Response: 01 04 04 00 00 00 00 [FB 84]

- (40001)Read Digital input counter for DI1~DI2 and return(00 00 00 00).
  Request : 01 03 00 01 00 02 [95 CB]
  Response: 01 03 04 00 00 00 00 [FA 33]

- (00513) Clear the DI2 counter value and return successful.
  Request : 01 05 02 02 FF 00 [2C 42]
  Response: 01 05 02 02 FF 00 [2C 42]

- (00513) Clear the DI counter value for DI0~DI3 and return successful.
  Request : 01 0F 02 00 00 04 01 0F [7F 70]
  Response: 01 0F 02 00 00 04 [55 B0]

- (02251) set DI1 input count to rising edge (1->0) and return successful.
  Request : 01 05 08 CB FF 00 [FF A4]
  Response: 01 05 08 CB FF 00 [FF A4]

5.5.7.5 (00129) Safe value for DO0~DO31

- (00129) Set output channel(0,2,4,5,9) to ON(0000 0010 0011 0101) for write safe value and return successful.
  Request : 01 0F 00 80 00 0F 02 35 02 [6C A5]
  Response: 01 0F 00 80 00 0F [14 27]

- (00129) Read safe value for output channel(0~12) and return safe value(35 02)
  Request : 01 01 00 80 00 0F [7D E6]
  Response: 01 01 02 35 02 [2F 6D]

5.5.7.6 (00161) Power-on value for DO0~DO31

- (00161) Set output channel(0,2,4,5,9) to ON(0000 0010 0011 0101) for write power-on value and return successful.
  Request : 01 0F 00 A0 00 0F 02 35 02 [6B C5]
  Response: 01 0F 00 A0 00 0F [15 ED]

- (00161) Read power-on value for output channel(0~12) and return value(35 02)
  Request : 01 01 00 A0 00 0F [7C 2C]
  Response: 01 01 02 35 02 [2F 6D]

5.5.7.7 (42201) Digital input/output LED Configuration(for 80xxD)

- (42201) Set Digital input/output LED to Turn-ON LED when output active(bit-1=0) and Turn-ON LED when input high(bit-0=1) and return successful.
  Request : 01 06 08 98 00 01 [CB 85]
  Response: 01 06 08 98 00 01 [CB 85]

- (42201) Read Digital input/output LED configuration and return(00 01) Turn-ON LED when output active and Turn-ON LED when input high.
  Request : 01 03 08 98 00 01 [07 85]
  Response: 01 03 02 00 01 [79 84]

- (42201) Set Digital input/output LED to turn-ON LED when input high(bit-1=1) and turn-ON LED when output inactive(bit-0=1) and return successful.
  Request : 01 10 08 98 00 01 02 00 03 [73 89]
  Response: 01 10 08 98 00 01 [82 46]

5.5.7.8 (42209) DIO active status

- (42209) Set DIO input value 1 for non-signal or the low voltage (bit-0=0) and output value 1 for output inactive (bit-1=1) and return successful.
  Request : 01 06 08 A0 00 02 [0A 49]
  Response: 01 06 08 A0 00 02 [0A 49]

- (42209) Read DIO active status and return(00 02) , output value 1 for output inactive(bit-1=1) and input value 1 for non-signal (bit-0=0).
  Request : 01 03 08 A0 00 01 [86 48]
  Response: 01 03 02 00 02 [39 85]

- (42209) Set input value 1 for high voltage, input value 0 for non-signal(bit-0=1) and output value 1 for output active (bit-1=0) and return successful.
  Request : 01 03 08 A0 00 01 [86 48]
  Response: 01 03 02 00 02 [39 85]

5.5.8 Example of modbus RTU sub-function codes commands

5.5.8.1 (0x46:0x00) read the name of a module

Modules		All EDAM modules
Description		This function is used to read the name of a module.
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x00 - Sub-Function code
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x00 - Sub-Function code
03~06	4 bytes	The module name, = 0x00 0x80 0x11 0x00 - for EDAM-8011D
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Read address(01) module name and return the moudule name 8043 (0x00 0x80 0x43 0x00)
  Request : 01 46 00 [12 60]
  Response: 01 46 00 00 80 43 00 [35 BE]

5.5.8.2 (0x46:0x04) Set the address of a module

Modules		All EDAM modules
Description		This function is used to set the address of a module.
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x04 - Sub-Function code
03	1 byte	= 0x01~0xF7(1~247) - New address
04~06	3 bytes	= 0x00 0x00 0x00 - Reserved
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x04 - Sub-Function code
03	1 byte	Set address result = 0x00 - OK, others: error
04~06	3 bytes	= 0x00 0x00 0x00 - Reserved
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Write address(01) module new address(02) and Return OK(0x00).
  Request : 01 46 04 02 00 00 00 [F5 1E]
  Response: 01 46 04 00 00 00 00 [F4 A6]

5.5.8.3 (0x46:0x05) Read the communication

Modules		All EDAM modules
Description		This function is used to read the communication.
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x05 - Sub-Function code
03	1 byte	= 0x00 - Reserved
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x05 - Sub-Function code
03	1 byte	= 0x00 - Reserved
04	1 byte	Baud Rate code(CC), Ref. baud setting of configuration table
05~07	3 bytes	= 0x00 0x00 0x00 - Reserved
08	1 byte	communication protocol Mode = 0 - ASCII format protocol = 1 - Modbus RTU protocol
09	1 byte	Reserved, = 0x00
10	1 byte	CRC check = 0 - Disable CRC check = 1 - Enable CRC check
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Read address(01) communication settings and Return 19200 baudrate(07) , Modbus RTU(01) mode and CRC enabled(01).
  Request : 01 46 05 00 [E3 5D]
  Response: 01 46 05 00 07 00 00 00 01 00 01 [39 43]

5.5.8.4 (0x46:0x06) Set the communication settings

Modules		All EDAM modules
Description		This function is used to Set the communication settings
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x06 - Sub-Function code
03	1 byte	= 0x00 - reserved
04	1 byte	Baud Rate code(CC), = 0x03 ~ 0x0A
05~07	3 bytes	= 0x00 0x00 0x00 - Reserved
08	1 byte	communication protocol Mode = 0 - ASCII format protocol = 1 - Modbus RTU protocol
09	1 byte	Reserved, = 0x00
10	1 byte	CRC check, =1 - Enable CRC check
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x06 - Sub-Function code
03	1 byte	= 0x00 - reserved
04	1 byte	Baud Rate code(CC), = 0x00 - OK, others: error
05~07	3 bytes	= 0x00 0x00 0x00 - Reserved
08	1 byte	communication protocol Mode, = 0x00 - OK, others: error
09	1 byte	= 0x00 - reserved
10	1 byte	CRC check setting, = 0x00 - OK, others: error
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Set address(01) module baud rate 19200(07), Modbus RTU(01) and CRC enabled(01).

  Request : 01 46 06 00 07 00 00 00 01 00 01 [2D B3]

  Response: 01 46 06 00 00 00 00 00 00 00 00 [CB 73]

5.5.8.5 (0x46:0x20) Read the firmware version

Modules		All EDAM modules
Description		This function is used to read the firmware of a module.
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x20 - Sub-Function code
03	1 byte	= 0x00 - reserved
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x20 - Sub-Function code
03	1 byte	Major version, = 0x00 ~ 0xFF
04	1 byte	Minor version, = 0x00 ~ 0xFF
05	1 byte	Build version, = 0x00 ~ 0xFF
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Read address(01) module firmware version, Return(0D 02 01) the module firmware version “D02.01”.

  Request : 01 46 20 00 [F9 CD]

  Response: 01 46 20 0D 02 01 [D2 A6]

5.5.8.6 (0x46:0x21) Set Digital Input Count Edge

Modules		All EDAM modules
Description		This function is used to Set Digital Input Count Edge.
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x21 - Sub-Function code
03~06	3 bytes	Digital Input Count Edge, The data in the bytes response are packed as 8 input channels per one byte,1st byte contains the DI0~DI7 and the second byte contains DI8~DI15,..... = 1 - Rising edge = 0 - falling edge
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x21 - Sub-Function code
03	1 byte	Set Digital Input Count Edge = 0 - OK, others: error
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Set address(01) module DI(0,3) to rising edge and Return OK(0x00).
  Request : 01 46 21 09 00 00 00 [BA FD]
  Response: 01 46 21 00 [F8 5D]

5.5.8.7 (0x46:0x22) Read Digital Input Count Edge

Modules		All EDAM modules
Description		Read Digital Input Count Edge
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x22 - Sub-Function code
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x22 - Sub-Function code
03~06	3 bytes	Digital Input Count Edge, The data in the bytes response are packed as 8 input channels per one byte,1st byte contains the DI0~DI7 and the second byte contains DI8~DI15,..... = 1 - Rising edge = 0 - falling edge
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Read address(01) module Digital Input Count Edge and Return (09 00 00 00) the DI(0,3) rising edge.
  Request : 01 46 22 [92 79]
  Response: 01 46 22 09 00 00 00 [FE FD]

5.5.8.8 (0x46:0x27) Set Power-On output value

Modules		All EDAM modules
Description		This function is used to Set Set Power-On output value
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x27 - Sub-Function code
03~06	3 bytes	Power-on output value, The data in the bytes response are packed as 8 output channels per one byte,1st byte contains the DI0~DI7 and the second byte contains DI8~DI15,..... 1= ON and 0= OFF
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x27 - Sub-Function code
03	1 byte	Set Power-on output value, = 0 - OK, others: error
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Set address(01) module DO(0,3) out ON and Return OK(0x00).
  Request : 01 46 27 09 00 00 00 [32 FD]
  Response: 01 46 27 00 [FB FD]

5.5.8.9 (0x46:0x28) Read Power-On output value

Modules		All EDAM modules
Description		Read Power-On output value
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x28 - Sub-Function code
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x28 - Sub-Function code
03~06	3 bytes	Power-on output value, The data in the bytes response are packed as 8 output channels per one byte,1st byte contains the DI0~DI7 and the second byte contains DI8~DI15,..... 1 = ON and 0 = OFF
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Read address(01) module power-on output value and Return(09 00 00 00) the DO(0,3) output ON.
  Request : 01 46 28 [12 7E]
  Response: 01 46 28 09 00 00 00 [66 FC]

5.5.8.10 (0x46:0x29) Set DI/O active status

Modules		All EDAM modules
Description		This function is used to Set DI/O active status
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x29 - Sub-Function code
03	1 byte	Set DIO active status, (see p63) bit(0) - DI active status (IAS): = 0 - input value 1 for non-signal or high(open).; input value 0 for low(GND). = 1 - input value 0 for non-signal or high(open).; input value 1 for low(GND). bit(1) - DO active status(OAS): = 0 - output value 1 for output relay active, output value 0 for output relay inactive. = 1- output value 0 for output relay active, output value 1 for output relay inactive. bit(2~7) = 0
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x29 - Sub-Function code
03	1 byte	DIO active status value, = 0 - OK, others: error
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Set address(01) module Set bit(0)= 0, bit(1)= 1 and Return OK(0x00).
  Request : 01 46 29 02 [7E 5C]
  Response: 01 46 29 00 [FF 9D]

5.5.8.11 (0x46:0x2A) Read DI/O active status

Modules		All EDAM modules
Description		Read DI/O active status
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x2A - Sub-Function code
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x2A - Sub-Function code
03	1 byte	DI/O active status, bit(0) - DI(IAS) active status = 0 - input value 1 for non-signal or the low voltage, =1 - input value 1 for high voltage, input value 0 for non-signal or the low voltage(default). bit(1) - DO(OAS) active status = 0 - output value 1 for output active, output value 0 for output inactive(default). = 1 - output value 1 for output inactive, output value 0 for output active. bit(2~7) = 0
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Read address(01) module DI/O active status,Return(02) the bit(0)= 0, bit(1)= 1.
  Request : 01 46 2A [93 BF]
  Response: 01 46 2A 02 [7E AC]

5.5.8.12 (0x46:0x36) Set the response delay time

Modules		All EDAM modules
Description		This function is used to Set the response delay time
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x36 - Sub-Function code
03	1 byte	Response delay time(0~30ms)
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x36 - Sub-Function code
03	1 byte	response delay time, = 0 - OK, others: error
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Set address(01) module response delay time to 29ms and Return OK(0x00).
  Request : 01 46 36 1D [37 A4]
  Response: 01 46 36 00 [F7 AD]

5.5.8.13 (0x46:0x35) Read the response delay time

Modules		All EDAM modules
Description		This function is used to read the response delay time
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x35 - Sub-Function code
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x35 - Sub-Function code
03	1 byte	Response delay time(0~30ms)
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Read address(01) module response delay time and Return 29ms(1D).
  Request : 01 46 35 [D2 77]
  Response: 01 46 35 1D [37 54]

5.5.8.14 (0x46:0x80) Set Digital input/output LED Configuration

Modules		All EDAM modules
Description		This function is used to Set the response delay time
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x80 - Sub-Function code
03	1 byte	Digital input/output LED Configuration(for 80xxD), bit(1) - for digital output LED control: = 0 - Turn-ON LED when output active = 1 - Trun-ON LED when output inactive bit(0) - for digital input LED control: = 0 - Turn-ON LED when input low(active) = 1 - Trun-ON LED when input high(inactive)
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x80 - Sub-Function code
03	1 byte	DIO LED Configuration, = 0 - OK, others: error
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Set address(01) module DIO LED Configuration bit(0)= 0, bit(1)= 1 and Return OK(0x00).

  Request : 01 46 80 02 [00 0C]
  Response: 01 46 80 00 [81 CD]

5.5.8.15 (0x46:0x81) Read Digital input/output LED Configuration

Modules		All EDAM modules
Description		This function is used to read Digital input/output LED Configuration
Request		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x81 - Sub-Function code
Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0x46 - Function code
02	1 byte	= 0x81 - Sub-Function code
03	1 byte	Digital input/output LED Configuration(for 80xxD), bit(1) - for digital output LED control: = 0 - Turn-ON LED when output active = 1 - Trun-ON LED when output inactive bit(0) - for digital input LED control: = 0 - Turn-ON LED when input low(active) = 1 - Trun-ON LED when input high(inactive)
Error Response		
Offset	Length	Description
00	1 byte	= 0x01~0xF7(1~247) - Address of the module
01	1 byte	= 0xC6 - Error function code
02	1 byte	Exception Code, = 0x01 - invalid function code = 0x02 - invalid data address = 0x03 - invalid data value.

Example:

- Read address(01) module Digital input/output LED Configuration and Return(02) Trun-ON LED when output inactive, Turn-ON LED when input low(active).

  Request : 01 46 81 [D2 00]
  Response: 01 46 81 02 [01 9C]

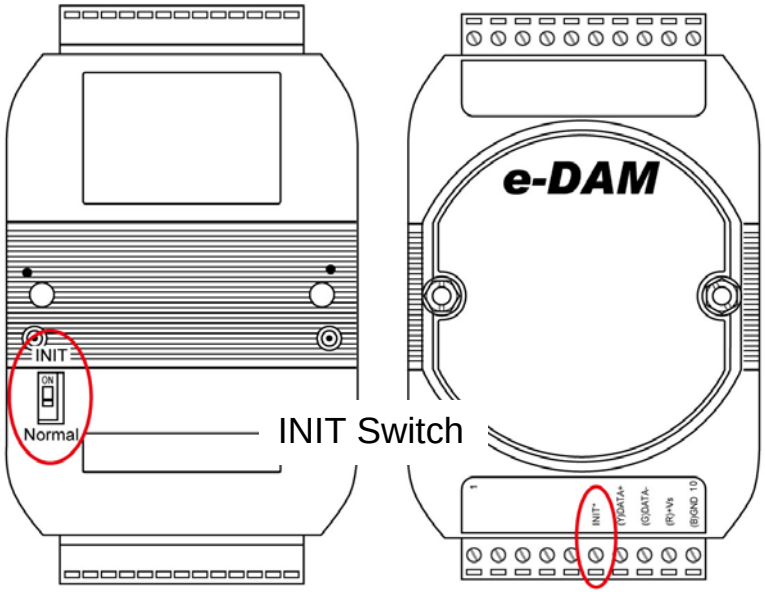
Appendix A INIT* pin(switch) operation

The "INIT*mode" has two purposes, one for reading module current configuration, and another for configuring the module baud rate and checksum.

■ Reading module current configuration

Each EDAM module has a built-in EEPROM which is used to store the configuration information such as address ID, type, baud rate etc.. If the user unfurtunally forget the configuration of the module. User may use a special mode called "INIT* mode" to resolve the problem When the module is set to "INIT* mode", the default settings are ID=00, baud rate=9600, and checksum=disable.

The INIT mode can be accessed by connecting the INIT* terminal to the GND terminal or by sliding the INIT switch to the Init position as shown below.



The following steps show you how to enable INIT* mode and read the current configuration:

1. Power off the module
2. Connect the "INIT*" pin to GND pin
   (or sliding the INIT switch to the Init position)
3. Power on the module
4. Send command $002(cr) in 9600 baud rate to read the current configuration stored in the EEPROM
5. Power off the module again
6. Open "INIT*" pin to force the module to normal mode

■ Configuring the module baud rate and checksum

The module should be set to "INIT* mode", While changing baud rate and/or checksum state by sending "Set module configuration" command (see section 3.13).

The following steps show you how to enable INIT* mode and change baud rate and/or checksum state

1. Power off the module
2. Connect the "INIT*" pin to GND pin
   (or sliding the INIT switch to the Init position)
3. Power on the module
4. Send command %AANNTTCCFF in 9600 baud rate to set baud rate and/or checksum state (*ID should be set to 00 in "INIT* mode"*)
5. Power off the module again
6. Open "INIT*" pin to force the module to normal mode
   (or sliding the INIT switch to the Normal position)

Appendix B Module Status

Power-On Reset or **Module Watchdog Reset** will let all output goto **Power-On Value**. And the module may accept the host's command to change the output value. Host Watchdog Timeout will let all digital output goto Safe Value. The host watchdog timeout flag is set, and the output command will be ignored. The module's LED will go to flash and user must reset the Module Status via command to restore normal operation.

Appendix C Dual Watchdog Operation

Dual Watchdog = Module Watchdog + Host Watchdog

The Module Watchdog is a hardware reset circuit to monitor the module's operating status. While working in harsh or noisy environment, the module may be down by the external signal. The circuit may let the module to work continues and never halt. The Host Watchdog is a software function to monitor the host's operating status. Its purpose is to prevent the network/communication from problem or host halt. While the timeout occurred, the module will turn the all output into safe state to prevent from unexpected problem of controlled target. The E-8000 module with Dual Watchdog may let the control system more reliable and stable.

Appendix D Reset Status

The reset status of a module is set when the module is powered-on or when the module is reset by the module watchdog. It is cleared after the responding of the first $AA5
command. This can be used to check whether the module had been reset. When the $AA5 command responds that the reset status is cleared, that means the module has not been reset since the last $AA5 command was sent. When the $AA5 command responds that the reset status is set and it is not the first time $AA5 command is sent, it means the module has been reset and the digital output value had been changed to
the power-on value.

Appendix E Input counter and Input latch

Input counter:

Each input channel has internal counter used to count the state change (*falling edge or rising edge* <see 3.13.2>) of input signal. The counting value can be read and cleared by sending "*Read digital input counter command*" or " *Clear digital input counter command*".(see 4.5.27)

Input latch:

Each input channel has internal latch which is used to latch the pulse signal from the input. This latched state can be read by sending "*Read latched digital input* " command and cleared by sending "*Clear latched digital input*" command.(see 4.5.32)

For example, if the digital input is connected to a key switch. The key switch is a pulse signal. The user may lose the strike information by sending command $AA6.

The digital input latch can latch the pulse and ready be read by sending "*Read latched digital input* " command. If the latched state=1 means that there is a key strike occurred.

Appendix F Power-on & Safe value

Power-on value:

Power-on value are used to set the module default output value when the module is turned-on or watch dog timeout reset. This function is especially importance in some application where the specified initial output states are required User can set power on value by sending *Set power-on/safe value* command *(*see section 4.5.40*)*

Safe value:

Safe value are used to set the module outputs into the specified values when Host watchdog timeout

If The host watchdog timer is enabled by sending *Set host watchdog timeout value*(see section 4.5.40), the host should send *Host OK* (see section 4.5.35) command periodically within Timeout value to refresh the timer, otherwise the module will be forced to safety state.

Appendix G Changing Configuration to Modbus Protocol

The EDAM-8000 Modbus version modules(firmware version D02.01 and later) may come from the factory set for which EDAM ASCII protocol are set as the default protocol. If the module is connected to a Modbus network, the Modbus network may not recognize the module. This may be caused by the incorrect settings. EDAM-8000 module should be set-up for Modbus protocol instead of EDAM ASCII protocol.
(see 4.5.9)

Please follow the steps as below for configuring an EDAM-8000 module to Modbus protocol.

1. Configure the EDAM-8000 Module with the EADAM-8000 utility
2. Initialize the EDAM-8000 on a RS-485 network (the preferred method is one module at a time on the RS-485 network).
3. With the module powered off, connect the INIT* terminal to the GND terminal.
4. Power up the module
5. Wait 10 seconds for the module to initialize.
6. Using the EDAM-8000 utility, search (scan) for the module to change the protocol. (Initial COM settings: 9600 baud, N-8-1)
7. The utility will identify the module from the search function.
8. The EDAM-8000 utility will now permit the serial data protocol to be changed to the Modbus protocol.
9. The address and COM port settings can also be changed at this time.
10. To access the module, click on the module icon in the utility.
11. Update the settings by pressing the “Update” button.
12. Power off the module.
13. Remove the wire between the INIT* and GND terminals
14. Power up the module
15. The module is now ready to be placed in the Modbus network.